TABLE OF CONTENTS

TABLE OF CONTENTS, CONTINUED

Hem-830-H-14

NAS 1.26: 27...

NASA CONTRACTOR REPORT

NASA CR-2729

NASA CR-2729

BOUNDARY-FITTED CURVILINEAR COORDINATE SYSTEMS FOR SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS ON FIELDS CONTAINING ANY NUMBER OF ARBITRARY TWO-DIMENSIONAL BODIES

Joe F. Thompson, Frank C. Thames, and C. Wayne Mastin

Prepared by
MISSISSIPPI STATE UNIVERSITY
Mississippi State, Miss. 39762
for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • JULY 1977

255

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-2729		

4. Title and Subtitle	5. Report Date
Boundary-Fitted Curvilinear Coordinate Systems For Solution of Partial Differential Equations on Fields Containing Any Number of Arbitrary Two-Dimensional Bodies	July 1977
	6. Performing Organization Code
7. Author(s) Joe F. Thompson, Frank C. Thames, and C. Wayne Mastin	8. Performing Organization Report No.
	10. Work Unit No.
9. Performing Organization Name and Address Mississippi State University Mississippi State, Mississippi 39762	11. Contract or Grant No. Grant NGR 25-001-055
	13. Type of Report and Period Covered Contractor Report
12. Sponsoring Agency Name and Address National Aeronautics & Space Administration Washington, DC 20546	14. Sponsoring Agency Code 3860

15. Supplementary Notes

Ruby Davis of the Theoretical Aerodynamics Branch of Langley Research Center converted the UNIVAC 1106 code generated at Mississippi State University to the CDC 6000 Series code at Langley Research Center. Percy J. Bobbitt was the Langley Technical Monitor. Final report.

16. Abstract

A method for automatic numerical generation of a general curvilinear coordinate system with coordinate lines coincident with all boundaries of a general multi-connected two-dimensional region containing any number of arbitrarily shaped bodies is presented. No restrictions are placed on the shape of the boundaries, which may even be time-dependent, and the approach is not restricted in principle to two dimensions. With this procedure the numerical solution of a partial differential system may be done on a fixed rectangular field with a square mesh with no interpolation required regardless of the shape of the physical boundaries, regardless of the spacing of the curvilinear coordinate lines in the physical field, and regardless of the movement of the coordinate system in the physical plane. A number of examples of coordinate systems and application thereof to the solution of partial differential equations are given. The FORTRAN computer program and instructions for use are included.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Coordinate transformation Multielement airfoils Navier-Stokes	Unclassified - Unlimited Subject Category 02

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	253	$9.00

*For sale by the National Technical Information Service, Springfield, Virginia 22161

Blank Page

BOUNDARY-FITTED CURVILINEAR COORDINATE SYSTEMS FOR SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS ON FIELDS CONTAINING ANY NUMBER OF ARBITRARY TWO-DIMENSIONAL BODIES

By Joe F. Thompson,† Frank C. Thames,* and C. Wayne Mastin§

Mississippi State University
Mississippi State, Mississippi 39762

Research Sponsored by NASA Langley Research Center, Grant NGR 25-001-055

SUMMARY

A method for automatic numerical generation of a general curvilinear coordinate system with coordinate lines coincident with all boundaries of a general multi-connected, two-dimensional region containing any number of arbitrarily shaped bodies is presented. No restrictions are placed on the shape of the boundaries, which may even be time-dependent, and the approach is not restricted in principle to two dimensions. With this procedure the numerical solution of a partial differential system may be done on a fixed rectangular field with a square mesh with no interpolation required regardless of the shape of the physical boundaries, regardless of the spacing of the curvilinear coordinate lines in the physical field, and regardless of the movement of the coordinate system in the physical plane. A number of examples of coordinate systems and application thereof to the solution of partial differential equations are given. The FORTRAN computer program and instructions for use are included.

I. INTRODUCTION

There arises in all fields concerned with the numerical solution of partial differential equations the need for accurate numerical representation of boundary conditions. Such representation is best accomplished when the boundary is such that it is coincident with some coordinate line, for then the boundary can be made to pass through the points of a finite difference grid constructed on the coordinate lines; hence the choice of cylindrical coordinates for circular boundaries, elliptic coordinates for

---

†Professor of Aerophysics and Aerospace Engineering, Ph.D.

*Engineering Specialist, Ph.D., Present Affiliation: LTV Aerospace Corporation, Dallas, Texas.

§Associate Professor of Mathematics, Ph.D.

elliptical boundaries, etc. Finite difference expressions at, and adjacent to, the boundary may then be applied using only grid points on the intersections of coordinate lines, without the need for any interpolation between points of the grid.

The avoidance of interpolation is particularly important for boundaries with strong curvature or slope discontinuities, both of which are common in physical applications. Likewise, interpolation between grid points not coincident with the boundaries is particularly inaccurate with differential systems that produce large gradients in the vicinity of the boundaries, and the character of the solution may be significantly altered in such cases. In most partial differential systems the boundary conditions are the dominant influence on the character of the solution, and the use of grid points not coincident with the boundaries thus places the most inaccurate difference representation in precisely the region of greatest sensitivity. The generation of a curvilinear coordinate system with coordinate lines coincident with all boundaries (herein called a "boundary-fitted coordinate system" for purposes of identification) is thus an essential part of a general numerical solution of a partial differential system.

A general method of generating boundary-fitted coordinate systems is to let the curvilinear coordinates be solutions of an elliptic partial differential system in the physical plane, with Dirichlet boundary conditions on all boundaries. One coordinate is specified to be constant on each of the boundaries, and a monotonic variation of the other coordinate around each boundary is specified. Thus, there is a coordinate line coincident with each boundary. The procedure is not restricted to two dimensions, allows the coordinate lines

to be concentrated as desired, and is applicable to all multi-connected regions (and thus to fields containing any number of arbitrarily shaped bodies).

The coordinate system so generated is not necessarily orthogonal, but orthogonality is not required, and its lack only requires that the partial differential system to be solved on the coordinate system when generated must be transformed directly through implicit partial differentiation rather than by use of the scale factors and differential operators developed for orthogonal curvilinear systems. An orthogonal system cannot be achieved with aribtrary spacing of the coordinate lines, and the capability for such concentration of coordinate lines is of more importance than orthogonality.

This general idea has been applied previously to two-dimensional regions interior to a closed boundary (simply-connected regions) by Winslow [1], Barfield [2], Chu [3], Amsden and Hirt [4], and Godunov and Prokopov [5]. Winslow [1] and Chu [3] took the transformed coordinates to be solutions of Laplace's equation in the physical plane which, as is shown in the next section, makes the physical cartesian coordinates solutions of a quasi-linear elliptic system in the transformed plane. Barfield [2] and Amsden and Hirt [4] reversed the procedure, taking the physical coordinates to be solutions in the transformed plane of a linear elliptic system which consists of Laplace's equation modified by a multiplicative constant on one term. This makes the transformed coordinates solutions of a quasi-linear elliptic system in the physical plane. Barfield also considered a hyperbolic system, but such a system cannot be used to treat general closed boundaries, since only elliptic systems allow specification of boundary conditions on the entirety of closed boundaries. Stadius [6] also used a hyperbolic system to generate a coordinate system for a doubly-connected region having parallel inner and outer boundaries.

With parallel boundaries it is only necessary to specify conditions on one of the boundaries, the location of the other boundary being free. The elliptic system, however, allows all boundaries to be specified as desired and thus has much greater flexibility.

Amsden and Hirt [4] constructed the coordinate generation method by iterative weighted averaging of the values of the physical coordinates at fixed points in the transformed plane in terms of values at neighboring points. Although not stated as such, this procedure is precisely equivalent to solving Laplace's equation, or modification thereof of the form noted above in Barfield [2], for the physical coordinates in the transformed plane by Gauss-Seidel iteration. Amsden and Hirt also allowed the boundary to move at each iteration, but this is simply equivalent to approaching the solution of the boundary-value problem through a succession of boundary-value problems converging to the problem of interest. In the approach of Godunov and Prokopov [5] the elliptic system is quasi-linear in both the physical and transformed planes. These authors applied a second transformation to that used by Chu [3], the transformation functions of this latter transformation being chosen a priori to control the coordinate spacing. Though not stated as such, the overall transformation may be shown to be generated by taking the transformed coordinates to be solutions in the physical plane of Laplace's equation modified by the addition of a multiple of the square of the Jacobian, the multiplicative factors being a priori chosen functions of the physical coordinates.

Meyder [7] generated an orthogonal curvilinear system by solving for the potential and "force" lines in a simply-connected region and taking these as the coordinate lines. This amounts to making the curvilinear coordinates

solutions of Laplace equations in the physical plane with Dirichlet boundary conditions (constant) on part of the boundary and Neumann boundary conditions (vanishing normal derivative) on the remainder. The solution for the coordinates was done, however, in the physical plane on a rectangular grid using interpolation at the curved boundaries, rather than in the transformed plane.

Orthogonal curvilinear coordinates for multi-connected regions, including regions with two bodies, have been generated by Ives [8] using conformal mapping. Conformal mapping is a special case of the generation of coordinate systems by solving an elliptic boundary value problem, but is not extendable to three dimensions and is less flexible in the spacing of the coordinate lines.

There have also been a number of transformations developed directly for special purposes without solving a partial differential system. One such approach is that of Gal-Chen and Somerville [9] for the treatment of an irregular boundary, such as mountaneous terrain, in a simply-connected region.

In the present research, the technique of generating the transformed coordinates as solutions of an elliptic differential system in the physical plane has been applied to multi-connected regions with any number of arbitrarily shaped bodies (or holes). The elliptic equations for the coordinates are solved in finite difference approximation by SOR iteration. Procedures for controlling the coordinate system so that coordinate lines can be concentrated as desired have been developed. Present effort is confined to two dimensions in the interest of computer economy, but the technique is extendable in principle to three dimensions. The procedure is also applicable

to fields with time-dependent boundaries, one coordinate line remaining fixed to the moving boundary. Here the equations for the coordinates must be re-solved at each time step. The computational grid remains fixed in spite of the movement of the physical grid.

Any partial differential system can be solved on the boundary-fitted coordinate system by transforming the set of partial differential equations of interest, and associated boundary conditions, to the curvilinear system. (It is shown in Appendix B that the equations do not change type, i.e., elliptic, parabolic, hyperbolic, under the transformation.) Since the boundary-fitted coordinate system has coordinate lines coincident with the surface contours of all bodies present, all boundary conditions can be expressed at grid points, and normal derivatives on the bodies can be represented using only finite differences between grid points on coordinate lines, without need of any interpolation, even though the coordinate system is not orthogonal at the boundary. The transformed equations can then be approximated using finite difference expressions and solved numerically in the transformed plane. Thus, regardless of the shape of the physical boundaries, and regardless of the spacing of the finite grid in the physical field, all computations, both to generate the coordinate system and, subsequently, to solve the partial differential system of interest can be done on a rectangular field with a square mesh with no interpolation required on the boundaries. Moreover, the physical boundaries may even be time-dependent without affecting the grid in the transformed region.

The computer software utilized to generate the boundary-fitted coordinate system is independent of the set of partial differential equations to be solved on this system. For example, numerical solutions for inviscid

and viscous fluid flows have been obtained using this system (Ref. 10-14). The partial differential equations governing these phenomena differ drastically. However, for a given body geometry, the same boundary-fitted system generation program was used in both solutions. Another major advantage of using boundary-fitted coordinates is that the computer software generated to approximate the solution of a given set of partial differential equations is completely independent of the physical geometry of the problem. The coordinate systems for the wide variety of bodies included in this report, for example, were all developed utilizing the same computer program. Finally, it is shown in Appendix C that physical integral conservation relations need not be lost in the transformed plane.

This report presents a detailed development of the method for generation of boundary-fitted coordinate systems for general, multi-connected, two-dimensional regions. The basic doubly-connected region transformation is discussed in Section II in some detail. Extensions of the basic transformation to multi-connected regions, contracted coordinate systems, and time-dependent systems are also discussed. The numerical techniques used to implement the method and a number of specific examples are presented in the Sections III and IV. Examples of application to the solution of partial differential equations are given in Section V. Finally, instructions for use and a listing of the FORTRAN program are given in Section VI. Various derivative relations used in the transformation of partial differential systems and program parameters used in the examples included are given in Appendix A.

II. MATHEMATICAL DEVELOPMENT

Preliminaries

The general transformation from the physical plane $[x,y]$ to the transformed plane $[\xi,\eta]$ is given by the vector-valued function

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} \xi(x,y) \\ \eta(x,y) \end{bmatrix} \tag{1}$$

The Jacobian matrix for this transformation is

$$\underline{J_1} = \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} \tag{2}$$

where the subscripts denote partial differentiation in the usual manner. The inverse function or transformation of (1) is, if it exists,

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x(\xi,\eta) \\ y(\xi,\eta) \end{bmatrix} \tag{3}$$

The Jacobian matrix of (3) will be denoted by $\underline{J_2}$ and is given by

$$\underline{J_2} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} \tag{4}$$

The Jacobian determinant, or Jacobian as it is normally called, is then

$$J = \det[\underline{J_2}] = x_\xi y_\eta - x_\eta y_\xi \tag{5}$$

The Jacobian matrices, (2) and (4), are related by

$$\underline{J_1} = [\underline{J_2}]^{-1} \tag{6}$$

which implies the relations

$$\xi_x = y_\eta/J, \quad \xi_y = -x_\eta/J, \quad \eta_x = -y_\xi/J, \quad \eta_y = x_\xi/J \qquad \text{(7a,b,c,d)}$$

Partial derivatives are transformed as follows:

$$f_x = \frac{\partial(f,y)}{\partial(\xi,\eta)} \bigg/ \frac{\partial(x,y)}{\partial(\xi,\eta)} = \frac{(y_\eta f_\xi - y_\xi f_\eta)}{J} \tag{8}$$

$$f_y = \frac{\partial(x,f)}{\partial(\xi,\eta)} \bigg/ \frac{\partial(x,y)}{\partial(\xi,\eta)} = \frac{(-x_\eta f_\xi + x_\xi f_\eta)}{J} \tag{9}$$

where f is some sufficiently differentiable function of x and y. Higher derivatives are obtained by repeated application of (8) and (9). A comprehensive set of transformed derivatives, operators, unit vectors, and other useful relations is given in Appendix A.

Sufficient conditions for the transformation described above to exist are given by the inverse function theorem (Ref. 15). In particular, if the component functions of (1) are continuously differentiable at a point, say (x_o, y_o), and the Jacobian matrix (2) is nonsingular at (x_o, y_o) then there exists a disk N_o about (x_o, y_o) such that the inverse function (3) exists and (6) holds for all [x,y] in N_o. It is readily apparent that the theorem guarantees existence only in a local fashion. For this reason component functions of (1) which possess even more desirable properties than those required by the inverse function theorem are sought.

Since the basic idea of the present transformation is to let the component functions of (1) be solutions of an elliptic Dirichlet boundary value problem, an obvious choice is to require that $\xi(x,y)$ and $\eta(x,y)$ be either harmonic, subharmonic, or superharmonic. Harmonic functions have continuous derivatives of all orders. Moreover, harmonic functions obey a maximum principle, which states that the maximum and minimum values of the function must occur on the boundaries of the region D. Thus, since no extrema occur within D, the first derivatives of the function will not simultaneously

vanishes in D, and hence the Jacobian J will not be zero due to the presence of an extremum. (Note that this merely removes one condition which may cause the Jacobian to vanish.) Further, the maximum principle guarantees uniqueness of the coordinate functions $\xi(x,y)$ and $\eta(x,y)$, (Ref. 16), and thus ensures that no overlapping of the boundaries will occur. Subharmonic and superharmonic functions are also continuously differentiable and obey a maximum principle. (The maximum principle is not as strong for these functions as it is for harmonic functions.) A more general discussion of the mathematical properties of the transformation is given in [17].

Doubly-Connected Region

Consider the transformation of a two-dimensional, doubly-connected region D bounded by two simple, closed, arbitrary contours onto a rectangular region D* as shown in Figure 1. (The basic transformation is discussed here assuming that the body contour and outer boundary are transformed, respectively, to the constant η-lines forming the bottom and top sides of the transformed region. The more general case of segmented body contours transforming to any side of the transformed region follows analogously and is discussed in later sections. The computer program allows the body contour(s) and outer boundary to be segmented and placed around the sides of the transformed plane in any manner desired.) Let Γ_1 map onto Γ_1^*, Γ_2 map onto Γ_2^*, Γ_3 onto Γ_3^*, and Γ_4 onto Γ_4^*. For identification purposes region D will be referred to as the physical plane, D* as the transformed plane, and Γ_1 as the body contour. Note that the transformed boundaries (Γ_1^* and Γ_2^*) are made constant coordinate lines (η-lines) in the transformed plane. The contours Γ_3 and Γ_4 which connect the contours Γ_1 and Γ_2 are coincident in the physical

plane and thus constitute re-entrant boundaries in the transformed plane.

In view of the closing remarks of the previous section, consider taking Laplace's equation as the generating elliptic system. That is, let $\xi(x,y)$ and $\eta(x,y)$ be harmonic in D. Then

$$\xi_{xx} + \xi_{yy} = 0 \tag{10a}$$

$$\eta_{xx} + \eta_{yy} = 0 \tag{10b}$$

with the Dirichlet boundary conditions

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} \xi_1(x,y) \\ \eta_1 \end{bmatrix}, \quad [x,y] \in \Gamma_1 \tag{10c}$$

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} \xi_2(x,y) \\ \eta_2 \end{bmatrix}, \quad [x,y] \in \Gamma_2 \tag{10d}$$

where η_1 and η_2 are different constants $(\eta_2 > \eta_1)$, and $\xi_1(x,y)$ and $\xi_2(x,y)$ are specified monotonic functions on Γ_1 and Γ_2, respectively, varying over the same range. The arbitrary curve joining Γ_1 and Γ_2 in the physical plane, which transforms to the right and left sides of the transformed plane, specifies a branch cut for the multiple-valued function $\xi(x,y)$. Thus, the values of the physical coordinate functions $x(\xi,\eta)$ and $y(\xi,\eta)$ are the same on Γ_3 as on Γ_4, and these functions and their derivatives are continuous from Γ_3 to Γ_4. Therefore, boundary conditions are neither required nor allowed on Γ_3 and Γ_4. (A graphic analog to the above ideas can be found in most complex variable texts where Riemann surfaces are discussed. For example, see Levinson and Redheffer, Ref. 16.).

Since it is desired to perform all numerical computations in the uniform rectangular transformed plane, the dependent and independent variables must be interchanged in (10). Use of equation (A.18) of Appendix A yields the coupled system

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = 0 \tag{11a}$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = 0 \tag{11b}$$

where

$$\alpha \equiv x_\eta^{\ 2} + y_\eta^{\ 2} \tag{11c}$$

$$\beta \equiv x_\xi x_\eta + y_\xi y_\eta \tag{11d}$$

$$\gamma \equiv x_\xi^{\ 2} + y_\xi^{\ 2} \tag{11e}$$

with the transformed boundary conditions

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f_1(\xi,\eta_1) \\ f_2(\xi,\eta_1) \end{bmatrix}, \quad [\xi,\eta_1]\epsilon\ \Gamma_1{}^* \tag{11f}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} g_1(\xi,\eta_2) \\ g_2(\xi,\eta_2) \end{bmatrix}, \quad [\xi,\eta_2]\epsilon\ \Gamma_2{}^* \tag{11g}$$

The functions $f_1(\xi,\eta_1)$, $f_2(\xi,\eta_1)$, $g_1(\xi,\eta_2)$, and $g_2(\xi,\eta_2)$ are specified by the known shape of the contours Γ_1 and Γ_2 and the specified distribution of ξ thereon. As noted, boundary data are neither required nor allowed along the re-entrant boundaries $\Gamma_3{}^*$ and $\Gamma_4{}^*$.

The system given by the equations of (11) is a quasi-linear elliptic system for the physical coordinate functions, $x(\xi,\eta)$ and $y(\xi,\eta)$, in the

transformed plane. Although this system is considerably more complex than that given by (10), the boundary conditions (11f,g) are specified on straight boundaries, and the coordinate spacing in the transformed plane is uniform. The boundary-fitted coordinate system generated by the solution to (11) has a constant η-line coincident with each boundary in the physical plane. The ξ=constant lines may be spaced as desired around the boundaries since the assignment of the ξ-values to the $[x,y]$ boundary points via the functions f_1, f_2, g_1, and g_2 in (11f,g) is arbitrary. (Numerically the discrete boundary values $[x_k,y_k]$ are transformed to equi-spaced discrete ξ_k-points on both boundaries.) Control of the radial spacing of the η=constant lines and of the incidence angle of the ξ=constant lines at the boundaries is accomplished by varying the generating elliptic system (i.e., the system of which $\xi(x,y)$ and $\eta(x,y)$ are solutions) as will be demonstrated in Section IV. As illustrated in Figure 1, the left and right boundaries of the transformed plane are re-entrant boundaries, which implies that both solutions, $x(\xi,\eta)$ and $y(\xi,\eta)$, are required to be periodic in the region $\{[\xi,\eta] \mid -\infty < \xi < \infty,$ $\eta_1 \leq \eta \leq \eta_2\}$.

Multiply-Connected Region

The basic ideas and procedures introduced in the preceding section can be extended to regions containing more than one body--that is, to general multi-connected or multi-body regions. One transformation for two bodies is illustrated in Figure 2. The bodies are connected with one arbitrary cut, with an additional cut joining one of the body contours to the outer boundary. The physical plane contours Γ_1 - Γ_8 map respectively onto the contours Γ_1^* - Γ_8^* in the transformed plane. Note that the body defined by the union of Γ_7 and Γ_8 is split into two segments (Γ_7^* and Γ_8^*), as is the cut joining

this body and the one defined by contour Γ_1. The η-coordinate is the same for both of the bodies and cut between them. Conversely, the cut defined by Γ_3 and Γ_4 in the physical plane is taken as a ξ=constant line in the transformed plane as before for the single body case. The outer boundary contour Γ_2 maps onto the upper boundary in the $[\xi,\eta]$ plane, becoming a constant η-line in the manner of the one-body transformation. In contrast to the one-body transformation, two re-entrant boundaries occur for this two-body transformation. The left and right vertical boundaries ($\Gamma_4{}^*$, $\Gamma_3{}^*$) appear as before. In addition a horizontal re-entrant segment due to the coincidence of Γ_5 and Γ_6 in the physical plane arises. The coordinate functions and the derivatives thereof are thus continuous across these re-entrant boundaries.

The boundary-fitted coordinates for the multi-body transformations are again determined by the solution of the set of equations (11) with the added boundary conditions

$$\begin{bmatrix} x \\ \\ y \end{bmatrix} = \begin{bmatrix} h_1(\xi,\eta_1) \\ \\ h_2(\xi,\eta_1) \end{bmatrix}, \quad [\xi,\eta_1] \in \Gamma_7{}^* \tag{11h}$$

$$\begin{bmatrix} x \\ \\ y \end{bmatrix} = \begin{bmatrix} q_1(\xi,\eta_1) \\ \\ q_2(\xi,\eta_1) \end{bmatrix}, \quad [\xi,\eta_1] \in \Gamma_8{}^* \tag{11i}$$

to define the additional body. Note that boundary conditions cannot be specified along the re-entrant boundaries defined by $\Gamma_3{}^*$, $\Gamma_4{}^*$, $\Gamma_5{}^*$, and $\Gamma_6{}^*$. As with the basic transformation all numerical computations both to generate the system and subsequently to utilize the coordinates for

solving a set of partial differential equations, are executed on a rectangular field with a uniform grid.

Simply-Connected Region

For a simply-connected region there are no bodies in the field and hence, no cuts in the physical plane and no re-entrant boundaries in the transformed plane. The single continuous boundary surrounding the physical field transforms to the entire rectangular boundary of the transformed field. The manner in which the physical boundary is split into four segments for placement on the four sides of the rectangular boundary in the transformed plane is a matter of choice.

Coordinate System Control

Control of the spacing of the coordinate lines on the body is easily accomplished, since the points on the body are input to the program. The spacing of the coordinate lines in the field, however, must be controlled by varying the elliptic generating system for the coordinates. One method of variation is to modify the Laplace equations (10) by adding inhomogeneous terms to the right sides so that the generating system becomes

$$\xi_{xx} + \xi_{yy} = P(\xi,\eta) \; , \; \eta_{xx} + \eta_{yy} = Q(\xi,\eta) \tag{12}$$

In the transformed plane these equations become

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} + J^2(Px_\xi + Qx_\eta) = 0 \tag{13a}$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} + J^2(Py_\xi + Qy_\eta) = 0 \tag{13b}$$

The effect of changing the functions $P(\xi,\eta)$ and $Q(\xi,\eta)$ on the coordinate system can be seen by examining the system (12). If $P \equiv Q \equiv 0$, the system reduces to the pair of Laplace equations used as the original generating system, i.e., Eq. (10). Let ξ', η' be the solutions of (10) with boundary values on D. Let ξ'', η'' be the solutions to system (12) with the same boundary values and with positive functions P and Q on the right hand side of the equations. Now ξ'' and η'' are subharmonic on D. Thus for any constant k, the $\xi'' = k$ coordinate line would be closer to $\xi'' = \xi_{max}$ than would the $\xi' = k$ line. The same relation holds for the $\eta'' = k$ and $\eta' = k$ coordinate lines. Now if all or part of the curve $\xi' = \xi_{max}$ is a branch cut in D, this branch cut will move in the direction of increasing ξ to meet the curve $\xi'' = \xi_{max}$ if the right side of the system (12) is increased from 0 to a positive function P. An analogous discussion can be made on the effect of negative functions P and Q on the generated curvilinear coordinate system.

Even though a change in P or Q in a subregion of D would change the coordinate lines throughout the entire region, the effect would certainly be more pronounced in the subregion. Also, the greater the change in P and Q, the greater the movement of the coordinate lines. By varying the sign and magnitude of $P(\xi,\eta)$ and $Q(\xi,\eta)$ for different values of ξ and η, considerable control can be exerted over the coordinate line spacing as will be seen from the examples that follow in Section IV.

One particularly effective procedure is to choose P and Q as exponential terms, so that the coordinates are generated as the solutions of

$$\xi_{xx} + \xi_{yy} = -\sum_{i=1}^{n} a_i \operatorname{sgn}(\xi - \xi_i)\exp(-c_i|\xi - \xi_i|)$$

$$- \sum_{j=1}^{m} b_j \operatorname{sgn}(\xi - \xi_j)\exp\left(-d_j\sqrt{(\xi - \xi_j)^2 + (\eta - \eta_j)^2}\right)$$

$$\equiv P(\xi,\eta) \tag{14a}$$

$$\eta_{xx} + \eta_{yy} = -\sum_{i=1}^{n} a_i \operatorname{sgn}(\eta - \eta_i)\exp(-c_i|\eta - \eta_i|)$$

$$- \sum_{j=1}^{m} b_j \operatorname{sgn}(\eta - \eta_j)\exp\left(-d_j\sqrt{(\xi - \xi_j)^2 + (\eta - \eta_j)^2}\right)$$

$$\equiv Q(\xi,\eta) \tag{14b}$$

where the positive amplitudes and decay factors are not necessarily the same in the two equations. Here the first terms have the effect of attracting the ξ = constant lines to the $\xi = \xi_i$ lines in Equation (14a), and attracting η = constant lines to the $\eta = \eta_i$ lines in Equation (14b). The second terms cause ξ = constant lines to be attracted to the points (ξ_j,η_j) in (14a), with similar effect on η = constant lines in (14b). No computational difficulties have been encountered because of the discontinuities in P and Q caused by the sgn function, which is defined by setting sgn(x) to be 1, 0, or -1 depending on whether x is positive, zero, or negative. Should problems arise in later applications, this function can be replaced by $\frac{2}{\pi}$ Arctan(nx), where n is a large positive integer. The sgn function was chosen to give the maximum control. Several examples of the use of coordinate system control are given in Section IV.

With the inclusion of the sgn function (or the arctan function) the equations (14a & b) for the curvilinear coordinates are no longer subharmonic or superharmonic, since the sgn function causes a sign change on the right side when the attraction is to lines or points not on the boundaries. It is possible, therefore, that too strong an attraction amplitude may cause the system to overlap and therefore be unusable. Many successful systems (all those included in the examples given herein) have been generated, however, using the above equations.

The use of the sign-changing sgn function is only necessary to cause attraction to both sides of a line or point in the field. Elimination of this function causes attraction on one side and repulsion on the other. If it is only desired to concentrate coordinate lines near one boundary, such as the body surface, then there is no need for the sign change, and the sgn function can be eliminated. In this case the equations are subharmonic or superharmonic, and a maximum principle is in effect to prevent overlap. Such a choice is provided for in the computer program.

The subject of coordinate system control is still very much under investigation, and other control functions are being evaluated. It is anticipated that new coordinate control packages will be made available for inclusion in the code when warranted. Of particular interest is the capability to cause a specified number of lines to fall within a certain physical region, such as a boundary layer. Another area of further investigation is the coupling of the coordinate equations with the partial differential system to be solved thereon so that the coordinate lines concentrate automatically in regions of high gradient.

Time-Dependent Coordinate Systems

If the coordinate system changes with time then the grid points move in the physical plane. Ordinarily such movement of the physical grid points would require interpolation among the grid points to produce values of the dependent variables at the new locations of the grid points. With the present method of coordinate system generation, however, it is possible to perform all computation on the fixed rectangular grid in the transformed plane without any interpolation no matter how the grid points move in the physical plane as time progresses. This occurs as follows:

Recall that the coordinate system is generated as the solution of some elliptic system with the values of the transformed coordinates $[\xi,\eta]$ specified on the boundaries in the physical plane, one of these coordinates being specified to be constant on the boundaries and the other being distributed as desired along the boundaries in order, perhaps, to concentrate grid points in certain regions. The transformed coordinates define a rectangular plane, the extent of which is determined by the range of the values of ξ and η. Now if the same boundary values of ξ and η are redistributed in the physical plane, perhaps because the boundaries in the physical plane have actually moved or maybe just to change the concentration of grid points around the boundaries, and the elliptic system is re-solved for the transformed coordinates with these new boundary conditions, new transformation functions can be produced with still the same range of values in ξ and η (provided the elliptic system used exhibits a maximum principle) and hence to the same rectangular field in the transformed plane. The grid points in the rectangular transformed plane thus remain stationary, and the effect of the movement of the coordinate system in the physical plane is just to change the values of the physical coordinates $[x,y]$ at the fixed grid points in the rectangular transformed plane.

Thus, although the position of a grid point changes on the physical plane, its position in the transformed plane is fixed. The time derivative transforms to the transformed plane as shown below:

$$\left(\frac{\partial f}{\partial t}\right)_{x,y} = \frac{\partial(x,y,f)}{\partial(\xi,\eta,t)} \bigg/ \frac{\partial(x,y,t)}{\partial(\xi,\eta,t)}$$

$$= f_t - x_t(f_\xi y_\eta - f_\eta y_\xi)/J$$

$$+ y_t(f_\xi x_\eta - f_\eta x_\xi)/J \qquad (15)$$

Here all derivatives are expressed in the transformed plane, so that the interpolation that would be necessary to supply values at grid points in the physical plane that have moved is not required in the transformed plane. (Note that in the transformed expression for the time derivative, all derivatives are taken at the fixed grid points in the transformed plane. The movement of the grid in the physical plane is reflected only through the rates of change of x and y at the fixed grid points in the transformed plane.)

The problem of solving N partial differential equations of any type in a physical region with time dependent boundaries has been replaced by a new problem consisting of N + 2 equations with fixed boundaries. The two additional equations are, of course, those governing the transformation (either (10) or (12)). Thus, it is possible to construct numerical solutions to physical problems with time dependent boundaries in a fixed rectangular plane with a fixed square mesh with no interpolation required. Again the above statements imply the independence of computer software from physical geometry. Problems involving moving blast fronts, shocks, free surfaces, and any other time-varying boundaries can be attacked successfully with these

procedures. (Some application to free surface flow is illustrated in Ref. 26.) In addition this method allows time-dependent concentration of grid points as desired in the physical plane.

The use of time-dependent coordinate systems requires that the difference equations for the curvilinear coordinates be resolved at each time step, of course. The coordinate values at the previous time step can serve as the initial guess for the next, however, so that the iteration will converge rapidly.

III. NUMERICAL SOLUTION

Difference Equations

The discussion here assumes the body contour and outer boundary transform, respectively, to the η-lines forming the lower and upper sides of the rectangular transformed plane as discussed in Section II. More general segmentation of the body contour(s) and the outer boundary, and placement as desired around the boundary of the transformed field, are provided for in the computer program.

The finite difference grid for the single-body problem is illustrated in Figure 3a. Circles denote the points at which the difference approximation to (11a,b) or (13a,b), are applied, while the triangles denote boundary points. The left and right vertical boundaries are coincident in the physical plane, and the values of x and y are thus equal along these lines. Such lines are designated re-entrant boundaries as indicated before. If the number of ξ = constant lines is designated IMAX and the number of η-lines by JMAX, the computational field size is (JMAX-2)(IMAX-1). Boundary values are specified

on j=1 and j=JMAX for all $1 \le i \le IMAX$. The j=1 line corresponds to the contour Γ_1 (the body contour) in the physical plane while j=JMAX is associated with the remote boundary contour Γ_2. Second-order central difference expressions (see Appendix D) are used to approximate all derivatives in the transformed equations. The resulting equations are, for (11a,b),

$$x_{i,j} = [\alpha'_{i,j}(x_{i-1,j} + x_{i+1,j}) - \beta'_{i,j}(x'_{\xi\eta})_{i,j}/2$$

$$+ \gamma'_{i,j}(x_{i,j+1} + x_{i,j-1})]/[2(\alpha'_{i,j} + \gamma'_{i,j})] \tag{16a}$$

$$y_{i,j} = [\alpha'_{i,j}(y_{i-1,j} + y_{i+1,j}) - \beta'_{i,j}(y'_{\xi\eta})_{i,j}/2$$

$$+ \gamma'_{i,j}(y_{i,j+1} + y_{i,j-1})]/[2(\alpha'_{i,j} + \gamma'_{i,j})] \tag{16b}$$

where $\alpha'_{i,j}$, $\beta'_{i,j}$, $\gamma'_{i,j}$, $(x'_{\xi\eta})_{i,j}$, and $(y'_{\xi\eta})_{i,j}$ are the difference approximations for α, β, γ, and the cross derivatives respectively. These expressions are developed in Appendix D. The re-entrant boundaries occurring at i=1 and i=IMAX are dealt with as follows. Since the values of x and y are equal along these lines, iteration is necessary along only one of them. Choosing i=1 for convenience, the ξ-derivatives along this line are approximated as exhibited below:

$$(x_\xi)_{1,j} = (x_{2,j} - x_{IMAX-1,j})/2 \tag{17a}$$

$$(x_{\xi\xi})_{1,j} = x_{2,j} - 2x_{1,j} + x_{IMAX-1,j} \tag{17b}$$

$$(x_{\xi\eta})_{1,j} = (x_{2,j+1} - x_{2,j-1} + x_{IMAX-1,j-1} - x_{IMAX-1,j+1})/4 \tag{17c}$$

for $2 \leq j \leq JMAX-1$. Similar expressions are used for the derivatives of y. The set of non-linear simultaneous difference equations produced by (16a,b) is solved by point SOR iteration.

Multiple-Body Fields

A sample mesh for a two-body transformation is shown in Figure 3b. As before, circles denote computational nodes and triangles the boundary points. Values defining the outer boundary contour Γ_2 (see Figure 2) are required for $1 \leq i \leq IMAX$ along the $j=JMAX$ line. Boundary values specifying the shape of the body contours Γ_1, Γ_7, and Γ_8 in the physical plane are required in segments along the $j=1$ line as follows:

$$\Gamma_1: \quad I2 \leq i \leq I3$$

$$\Gamma_7: \quad 1 \leq i \leq I1$$

$$\Gamma_8: \quad I4 \leq i \leq IMAX$$

The above requirements may be more clearly seen by comparing Figures 2 and 3b. The difference equations (16a,b) and the vertical re-entrant boundary relations (17a,b,c) are also valid for multi-body transformations. The primary difference in the two transformations results from the existence, in the multi-body case, of the horizontal re-entrant boundaries along $j=1$. The equivalence of x and y values along these segments is demonstrated in Figure 3b. Again it is only required to calculate values along one re-entrant segment. Choosing the leftmost ($I1 \leq i \leq I2$), the η-derivatives are calculated using special procedures which are illustrated by the expressions below for the point marked with an ⊗ ($i,j = I1 + 1,1$) in Figure 3b:

$$(x_\eta)_{I1+1,1} = (x_{I1+1,2} - x_{I4-1,2})/2 \tag{18a}$$

$$(x_{\eta\eta})_{I1+1,1} = x_{I1+1,2} - 2x_{I1+1,1} + x_{I4-1,2} \tag{18b}$$

$$(x_{\xi\eta})_{I1+1,1} = (x_{I4,2} - x_{I4-2,2} + x_{I1+2,2} - x_{I1,2})/4 \tag{18c}$$

Note the existence of the horizontal re-entrant boundaries increases the size of the computational field somewhat. In the two-body example given the number of additional equations to be solved is I2 - (I1+1).

Multiple-Body Segment Arrangements

In the case of a single body it is logical to keep the body contour in one segment, with a single cut connecting the single segment to the outer boundary. This type of arrangement is illustrated in Figure 4a. (Figure 4b shows an alternate single body arrangement. In these and all subsequent figures the dotted lines on the segment arrangement diagrams identify the two members of a re-entrant pair.) In the case of multiple bodies there is a wider choice of reasonable arrangements, some of which may be better than others for certain applications. The boundaries in the physical plane may be split into as many segments as desired, and these segments may be arranged around the rectangular boundary of the transformed plane in any way desired. These segments are all connected by branch cuts in the physical plane and by re-entrant boundaries in the transformed plane. Several of these arrangements are illustrated in Figures 5-13. Illustrative values of the segment input parameters are given in Table 1 for each of these arrangements, and input instructions are given in Section VI.

In the arrangement of Figure 5, an η-line encircles both bodies and forms a cut between the bodies, the cut to the outer boundary being a ξ-line. The outer boundary is also a line of constant η but at a different value. Here one body is split into two segments, while the other body and the outer boundary are each in single segments. Figure 6 shows an arrangement in which each body is in a single segment, each body being a ξ-line of different value. Here there is no cut between the bodies, but rather an η-line cut between each body and the outer boundary, which is split into two segments, each being an η-line of different value. (This produces a system similar to a bi-polar coordinate system.) In Figure 7 each body is also in a single segment with the outer boundary split into two segments, but here an η-line encircles each body and forms the cut between that body and the outer boundary. In Figure 8 one body is a single segment encircled by an η-line which forms a cut to the other body. The other body is split into two segments, each being a ξ-line of different value, with each segment connected to the outer boundary by a ξ-line. The outer boundary is in a single segment and is an η-line. Other arrangements are shown in Figures 9-13. All these arrangements are shown without coordinate line attraction, and, consequently, many of the resulting systems exhibit wide spacing in concave areas. This spacing can be improved by coordinate attraction as illustrated in the examples of Section IV. The results of the use of several of these multiple-body segment arrangements are given for two-body potential flow in Section V. Coordinate system control was used effectively in that case to improve the spacing in the concave region. These concave regions occur when he cut and body are on the same coordinate line.

Initial Guess

Since the difference equations are nonlinear, the initial guess must be within a certain neighborhood of the solution if the iterative solution is to converge. With some segment arrangements a logical choice of an initial guess is difficult to perceive. Therefore several different types of initial guesses have been inserted in the program, with the choice to be made by the user as guided by past experience. The rationale for some of these guesses is more intuitive than analytical. The choices available are detailed below. (In each case the initial values of x and y on all cuts are interpolated linearly between the boundary values at the cut end points.) The choice is controlled by the input parameter IGES as discussed in Section VI. The guess type is identified by this number in the discussion below.

(a) Weighted Average of Four Boundary Points - Here the values of x and y at each point in the field are set equal to the average of the four boundary points having either the same ξ index or the same η index, the average being weighted by the distance to the boundary in the transformed plane. Thus

$$2\, x_{ij} = \left(\frac{JMAX - j}{JMAX - 1}\right)x_{i,1} + \left(\frac{j - 1}{JMAX - 1}\right)x_{i,\,JMAX}$$

$$+ \left(\frac{IMAX - i}{IMAX - 1}\right)x_{1,j} + \left(\frac{i - 1}{IMAX - 1}\right)x_{IMAX,j} \qquad (19)$$

for i = 2, 3, ..., IMAX-1 and j = 2, 3, ..., JMAX-1. An analogous equation is used for y. (IGES = 1). A variation of this type (and also of types (b),(c), & (e)) is provided by IGED, whereby the average may be restricted to only a two-point average in either the ξ or η direction. The direction chosen should be that which proceeds between the bodies and the outer boundary. This variation is useful with an outer boundary located on three sides.

(b) Same as (a), except zeroes replace the values on the cuts in the formation of the average. This type of initial guess is particularly effective with simply-connected regions, single-body fields, and multiple-body segment arrangements having the all body segments on one horizontal (vertical) side and the outer boundary a single segment on the other horizontal (vertical) side. (IGES = 0)

(c) Weighted Average of Body Segment Boundary Points Only - This type guess is the same as that of (a), except that boundary points on cuts are not included in the formation of the average, which therefore may be formed with fewer than four points. This type and the exponential projection below are widely effective. (IGES = 2)

(d) Moment Projection - Here the initial value at each field point is given by

$$x_{ij} = \frac{(\sum_k d_{ijk})(\sum_k x_k) - \sum_k d_{ijk}x_k}{N \sum_k d_{ijk}} \tag{20}$$

$$\text{with } d_{ijk} = \sqrt{(x_{ij} - x_k)^2 + (y_{ij} - y_k)^2}$$

Here x_k is the value at a point on the boundary of the transformed plane; d_{ijk} is the distnace to that boundary point in the transformed plane; N is the total number of boundary points; and the summations extend over the entire boundary in the transformed plane (IGES = 3). A modification of this type omits the division by N (IGES = 4).

(e) Exponential Weighted Average of Body Segment Boundary Points - This guess is similar to that of (c) except that the weight in the average is exponential rather than linear. Increasing IGES causes the points to contract nearer the boundary segments corresponding

to the lowest values of η and ξ. This type is most effective when strong coordinate attraction is used with a single-body field having the body located on the bottom or left side of the rectangular transformed field. (IGES > 4)

(f.) Exponential Projection - The initial value of x is determined at each field point by

$$x_{ij} = \frac{\sum_k x_k \exp(-|IGES| d_{ijk}/d_o)}{\sum_k \exp(-|IGES| d_{ijk}/d_o)} \tag{21}$$

where d_o is the diagonal length of the transformed field, the other quantities having the same definitions as in (d) above. (IGES < 0)

Examples of these initial guess types are shown in Figures 14-23 for the segment arrangements given in Figures 4-13. The value of IGES for each is given in the upper left corner of each plot. Table 2 lists the types for which convergence was obtained with each of these segment arrangements.

The most widely effective initial guess in these cases was the exponential projection. This type of guess produced convergence in the single-body case and in six of the nine two-body cases. The optimum decay factor for the exponential projection was 40 (IGES = -40) in most cases, with an optimum of 20 (IGES = -20) in a few cases.

Guess Type 2 (IGES = 2) also gave convergence in six of the nine two-body cases and in the single-body case. However, the number of iterations required was a bit larger than with the exponential projection. Type 2 gave convergence with one segment arrangement (Figure 9) for which the exponential projection gave divergence, but gave divergence for another arrangement (Figure 12) for which the exponential projection gave convergence.

Arrangements having the outer boundary in two segments tend to be the more difficult to converge. Of the four arrangements of this type (Figures

6, 7, 12, 13), Guess Type 2 gave convergence for only one (Figure 13), while the exponential projection failed for two of the four. The two arrangements of Figures 6 and 7 proved to be particular recalcitrant, requiring a switch to a different size field in order to get convergence for the arrangement of Figure 6 and the introduction of a special guess for that of Figure 7.

The general suggestion for two-body cases is to use either exponential projection with a decay factor of about 40 or Guess Type 2. For simply-connected regions, single-body cases, and the two-body segment arrangements having both bodies on the same coordinate line, Guess Type 0 is generally more efficient, although the exponential weighted average or projection and Guess Type 2 will also give convergence. Arrangements having two bodies in single segments on opposite sides of the transformed plane are particularly difficult, and Guesses Type 1 and 4 may be used.

With some segment arrangements with multiple bodies, convergence can be achieved with a close outer boundary, but not with the outer boundary farther out. Therefore, provision has been made for initially converging the solution with a circular outer boundary close in and then constructing an initial guess for a field with a larger circular outer boundary from this solution by linear projection to the larger field. This process can be repeated as many times as desired with the outer boundary gradually being moved out to its desired position. Two types of movement are provided: (a) the outer boundary radius is doubled at each step, or (b) the outer boundary radius is increased linearly at each step. This provision should not be used unless necessary, and then the movement of (a) is to be preferred in general, with as few steps as will produce convergence.

Finally, with strong coordinate line attraction it may not be possible to achieve convergence from an initial guess that gives convergence without attraction. Provision therefore has been made whereby the attraction can be

added gradually, the converged solution for a small attraction becoming the initial guess for a case of stronger attraction. Two types of increase in attraction strength are provided: (a) the attraction amplitude is doubled at each step, or (b) the attraction amplitude is increased linearly at each step. This procedure is to be used only when necessary. In general, type (a) is preferred, with as few steps as will provide convergence. Further discussion of the use of the various initial guesses is given in the instructions of Section VI.

Convergence Acceleration

For a difference equation of the general form

$$a_1(f_{i+1,j} + f_{i-1,j}) + a_2(f_{i,j+1} + f_{i,j-1}) + b_1(f_{i+1,j} - f_{i-1,j})$$
$$+ b_2(f_{i,j+1} - f_{i,j-1}) + cf_{ij} + d_{ij} = 0 \tag{22}$$

$$(i = 1, 2, --, I \; ; \; j = 1, 2, --, J)$$

with boundary values specified on $i = j = 0$, $i = I + 1$, and $j = J + 1$, and a_1, a_2, b_1, b_2, c, and d constant, the optimum value of the SOR acceleration parameter ω can be obtained in the case where $a_1^2 \leq b_1^2$ and $a_2^2 \leq b_2^2$, and in the case where $a_1^2 \geq b_1^2$ and $a_2^2 \geq b_2^2$, (Ref. 18). The optimum parameters in these two cases are as follows:

<u>Case #1: $a_1^2 \geq b_1^2$ and $a_2^2 \geq b_2^2$</u>

$$\omega = \frac{2}{1 + \sqrt{1 - \rho^2}} \quad \text{(over-relaxation, } 1 \leq \omega < 2) \tag{23}$$

<u>Case #2: $a_1^2 \leq b_1^2$ and $a_2^2 \leq b_2^2$</u>

$$\omega = \frac{2}{1 + \sqrt{1 + \rho^2}} \quad \text{(under-relaxation, } 0 < \omega \leq 1) \tag{24}$$

where

$$\rho = 2\sqrt{\left|\left(\frac{a_1}{c}\right)^2 - \left(\frac{b_1}{c}\right)^2\right|}\cos\frac{\pi}{I+1}$$

$$+ 2\sqrt{\left|\left(\frac{a_2}{c}\right)^2 - \left(\frac{b_2}{c}\right)^2\right|}\cos\frac{\pi}{J+1} \tag{25}$$

In the remaining case where $a_1^2 \gtrless b_1^2$ and $a_2^2 \lessgtr b_2^2$, no theoretical determination of the optimum acceleration parameter exists as yet.

Since the difference equations for the coordinate system are nonlinear, the above theory is not directly applicable. However, if the equations are considered as locally linearized then a local optimum acceleration parameter can be obtained which will vary over the field. It should be noted that the local linearization is applied only to the determination of the acceleration parameters, not to the actual solution of the difference equations.

Following this approach and neglecting the effect of the cross derivatives, the local constants in the above equations become

$$a_1 = \alpha \qquad\qquad a_2 = \gamma$$

$$b_1 = \frac{J^2P}{2} \qquad\qquad b_2 = \frac{J^2Q}{2}$$

$$c = -2(\alpha + \gamma)$$

so that locally optimum acceleration parameters are

Case #1: $\alpha_{ij} \geq \dfrac{J_{ij}^{\,2}|P_{ij}|}{2}$ and $\gamma_{ij} \geq \dfrac{J_{ij}^{\,2}|Q_{ij}|}{2}$

$$\omega_{ij} = \frac{2}{1 + \sqrt{1 - \rho_{ij}^{\,2}}} \qquad \text{(over-relaxation)} \tag{26}$$

Case #2: $\alpha_{ij} \le \frac{J_{ij}^2|P_{ij}|}{2}$ and $\gamma_{ij} \le \frac{J_{ij}^2|Q_{ij}|}{2}$

$$\omega_{ij} = \frac{2}{1 + \sqrt{1 + \rho_{ij}^2}} \quad \text{(under-relaxation)} \tag{27}$$

where

$$\rho_{ij} = \frac{1}{\alpha_{ij} + \gamma_{ij}}\left[\sqrt{\left|\alpha_{ij}^2 - \frac{J_{ij}^4 P_{ij}^2}{4}\right|}\cos\left(\frac{\pi}{IMAX - 1}\right) + \sqrt{\left|\gamma_{ij}^2 - \frac{J_{ij}^4 Q_{ij}^2}{4}\right|}\cos\left(\frac{\pi}{JMAX - 1}\right)\right] \tag{28}$$

In the remaining case where $\alpha \gtrless \frac{J^2|P|}{2}$ and $\gamma \lessgtr \frac{J^2|Q|}{2}$, not even a local optimum is available. The program allows a choice of strategy in this case: over-relaxation, under-relaxation, or a weighted average as follows:

First ρ_1 and ρ_2 are calculated from

$$\rho_1 = \frac{1}{\alpha + \gamma}\sqrt{\left|\alpha^2 - \frac{J^4P^2}{4}\right|}\cos\left(\frac{\pi}{IMAX - 1}\right) \tag{29a}$$

$$\rho_2 = \frac{1}{\alpha + \gamma}\sqrt{\left|\gamma^2 - \frac{J^4Q^2}{4}\right|}\cos\left(\frac{\pi}{JMAX - 1}\right) \tag{29b}$$

If over-relaxation is specified then ω is calculated from Eq. (26) using $\rho = \rho_1 + \rho_2$. The same expression for ρ is used in Eq. (27) if under-relaxation is specified. If the weighted average is to be used then, using ρ_1 in place of ρ, ω_1 is calculated from Eq. (26) if $\alpha \ge \frac{J^2|P|}{2}$ or by Eq. (27) if $\alpha \le \frac{J^2|P|}{2}$. Similarly, using ρ_2 in place of ρ, ω_2 is calculated from Eq. (26) if $\gamma > \frac{J^2|Q|}{2}$, else by Eq. (27). The average is then formed by

$$\omega = \frac{\rho_1\omega_1 + \rho_2\omega_2}{\rho_1 + \rho_2} \tag{30}$$

Optimum Acceleration Parameters

In order to provide some guide to the selection of acceleration parameters for the most rapid convergence of the iterative solution, the optimum values were determined in a number of representative cases by computer experimentation.

The body for this study was a Karman-Trefftz airfoil, the contour of which is shown in Figure 24. The points on the contour were spaced at equal angular increments in the complex plane from which the airfoil was generated, with three additional points added at half, fourth, and eighth angular increments above and below the trailing edge to provide finer resolution in that region.

A basic case was selected and four quantities, (the number of points on the body, the number of coordinate lines surrounding the body, the amplitude of the coordinate line attraction to the body, and the radius of the circular outer boundary) were varied individually and in pairs above and below the basic values. The initial guess type (Type 0) giving the fastest convergence for the basic case was used for all. Each case was run to convergence of 10^{-4}. Additional cases were run with different convergence criteria and different numbers of steps in the addition of the final attraction amplitude. The basic case was also run with a circular cylinder as the body for comparison of the effect of the body shape.

A series of two-body cases was also run with two of the same Karman-Trefftz airfoils positioned as an airfoil and flap system. Only one segment arrangement was considered, with both bodies on the same curvilinear coordinate line. The multiple-airfoil system and the segment arrangement are shown in Figure 25. The same set of quantities varied in the single body case was varied individually above and below the basic values (except

that variation of the number of points on the bodies above the basic value involved too much core and was omitted). The initial guess type was the same as that used for the single-body studies, which proved to give the fastest convergence in the basic two-body case as well. The values of all input parameters for the cases run are given in Tables 3-6.

The results of these studies are given in Tables 7-10. For the single-body field, Table 7 shows the effect of individual variation of each of the chosen quantities; Table 8 gives the effect of variation in pairs, and in Table 9 other miscellaneous quantities are varied. Table 10 gives the results for the double-body field. The number of iterations required with the variable acceleration parameter field and the average variable acceleration parameter over the field are also included in these tables. (Under-relaxation was used in the case of complex eigenvalues.) In a number of cases the variable acceleration parameters tend to be too large, and only in a few cases was the variable field better than the uniform experimentally determined optimum.

Plots of the number of iterations required vs. the acceleration parameter are given for a few cases in Figure 26. In a number of cases the optimum parameter was only 0.1 below the divergence limit for the particular case. A typical example of this appears in Figure 26b. The effect of the general field size is evident in Figure 26c and d, where the larger field (d) gives a much sharper minimum. Strong attraction with small fields makes the convergence more difficult as can be seen in Figures 26e and f. With strong attraction and a small field convergence was obtained only in a very narrow band of acceleration parameters.

A few trends are evident in these results:

(1) The optimum acceleration parameter, ω^*, and the number of iterations, I^*, increase with the number of grid points.

(2) ω^* increases slightly as the outer boundary moves outward, this effect being more pronounced at small outer boundary radius. I^* experiences a minimum as the outer boundary moves outward. Both of these effects are stronger with two bodies than with one.

(3) ω^* is little affected by attraction amplitude with one body, but tends to decrease with increasing attraction amplitude with two bodies. (This difference is probably due to the fact that in the two-body case there was attraction to a line in the field, i. e., the cut between the bodies, as well as the body contours.) I^* tends to increase with increasing attraction.

(4) There is an optimum number of steps for addition of the attraction, with increasing I^* occurring on both sides of this optimum. Too few steps may produce divergence. This optimum is less apparent with two bodies than with one.

(5) The number of attraction lines had only a small effect on either ω^* or I^* in the cases considered.

(6) ω^* increases as convergence tolerance is tightened.

(7) The intermediate convergence tolerance value that should be used when the attraction is added gradually should be 0.01. A tighter tolerance requires more iterations, while a looser tolerance may not produce a sufficiently close initial guess for the next addition of attraction.

(8) The use of the variable acceleration parameter field is not generally recommended in cases where the optimum can be estimated from experience. This feature can be useful, however, in the absence of a good estimate. In that

case it is probably best to use the variable field once, and then to use a factor about 3% less than the average over the variable for subsequent runs.

IV. EXAMPLES OF COORDINATE SYSTEMS

A variety of results utilizing the theory and numerical procedures outlined in previous sections are now presented. The results selected for presentation here were chose. to exemplify the generality of the method, and some were used for the numerical fluids studies in Ref. 11. In most of the plots only a portion of the physical field is shown in order that the coordinate lines may be distinguishable near the bodies. The actual fields were generally extended some ten chords in radius from the bodies.

Coordinate System Control

As discussed in Section II, the curvilinear coordinate lines may be concentrated by attracting the lines to other lines or points in the field. The control of the coordinate system in this manner is illustrated in Figures 27-28. Input parameters involved are given in Table 11. In Figure 27, the basic system generated by the Laplace equations (zero right hand sides) is shown in (a). In (b) the η-lines have been attracted to the body. In (c) the attraction to the body has been made stronger on two sides, while in (d) the lines are more strongly attracted over a small portion of the body. In (e) and (f) the angle of intersection of the lines with the body has been controlled, over the entire body in (e) and over only a portion of the body in (f).

Figure 28 illustrates the use of control to pull the coordinate lines into a concave portion of the body contour, (a) being the result of the Laplace equations and (b) having the lines attracted to the slope discontinuity on the lower surface.

Various Body Shapes

Coordinate systems for a circular cylinder and a cambered Joukowski airfoil are pictured in Figure 29. The contours are of equi-spaced ξ = constant and η = constant lines in the physical plane. (In the interest of clarity only a portion of the grid is shown in this and subsequent figures. The outer boundary for these and all succeeding results is circular and has a radius of ten body-lengths.) Note that the two systems given in Figure 29 are orthogonal (The transformations in these instances are conformal.). Slightly more general bodies are shown in Figure 30. These include a cambered and an integral flap Karman-Trefftz airfoil. Although systems for each of these could be generated by conformal transformations, the ones shown were obviously not. The effect of the coordinate system control is demonstrated for both airfoils in Figure 31. The figures exhibit the effect of contraction to the η-line coincident with the body profile. Note that the grid spacing is significantly collapsed in the contracted transformation. The contracted mesh spacing near the solid body boundary allows the solution of viscous flows (Ref. 11).

Contracted coordinate systems for more general airfoils are exhibited in Figures 32-34. These are the Liebeck laminar airfoil, the Göttingen 625 airfoil and a NACA 0018 profile. Contraction to the single-body η-line is demonstrated for the Liebeck profile and to the initial 15 η-lines for the Göttingen 625 and NACA 0018 airfoils. The effects of multiple-η-line contraction are seen to be quite dramatic.

Coordinate systems for a multiple-airfoil system are shown in Figure 35, with coordinate line concentration to the bodies and into the concave region formed by the airfoils and the cut between. The coordinate line attraction is to the first ten η-lines surrounding the bodies with an amplitude of 10,000

and a decay factor of 1.0 on all but the tenth line, where 0.5 was used. The coordinate system for simply connected regions are shown in Figures 36 and 37.

Finally, to demonstrate the applicability of the transformation method to quite arbitrary bodies, a system in contracted form (15 line) for a rather odd looking body--denoted the cambered rock--is given in Figure 38.

V. EXAMPLES OF APPLICATION TO PARTIAL DIFFERENTIAL EQUATIONS

As noted above, any set of partial differential equations may be solved on the boundary-fitted coordinate system by transforming the equations and associated boundary conditions and solving the transformed equations numerically in the transformed plane. All computation can be done on the fixed square grid in the rectangular transformed region regardless of the shape of the physical boundaries. Several examples of such application are given below.

Potential Flow (Ref. 11, 12, 19)

The two-dimensional irrotational flow about any number of bodies may be described by the Laplace equation for the stream function ψ:

$$\psi_{xx} + \psi_{yy} = 0 \tag{31}$$

with boundary conditions

$$\psi(x,y) = \psi_k \text{ on the surface of the kth body.} \tag{32a}$$

$$\psi(x,y) = y\cos\theta - x\sin\theta \text{ at infinity.} \tag{32b}$$

where θ is the angle of attack of the free stream relative to the positive x-axis. Here the stream function is nondimensionalized relative to the air-

foil chord and the free stream velocity. When transformed to the curvilinear coordinate system this equation becomes

$$\alpha\psi_{\xi\xi} - 2\beta\psi_{\xi\eta} + \gamma\psi_{\eta\eta} + \sigma\psi_{\eta} + \tau\psi_{\xi} = 0 \tag{33}$$

The transformed boundary conditions are (see Figure 1).

$$\psi(\xi,\eta_1) = \psi_o \text{ on } \eta = \eta_1 \text{ (i.e., on } \Gamma_1{}^*) \tag{34a}$$

$$\psi(\xi,\eta_2) = y(\xi,\eta_2)\cos\theta - x(\xi,\eta_2)\sin\theta \text{ on } \eta = \eta_2 \text{ (i.e., on } \Gamma_2^*) \tag{34b}$$

The uniqueness is implied by insisting that the solution be periodic in $-\infty < \xi < \infty$, $\eta_1 \leq \eta \leq \eta_2$. The coefficients α, β, γ, σ, and τ are calculated during the generation of the coordinate system (see Appendix A). Equation (33) was approximated using second-order, central differences for all derivatives, and the resulting difference equation was solved by accelerated Gauss-Seidel (SOR) iteration on the rectangular transformed field. The value of the boundary values of ψ on the bodies were determined by imposing the Kutta condition on each body.

The pressure coefficient at any point in the field may be obtained from the velocities via the Bernoulli equation, which in the present non-dimensional variables is

$$C_p = 1 - |\underset{\sim}{V}|^2 \tag{35}$$

On the body surface this becomes

$$C_p = 1 - \frac{\gamma}{J^2}\psi_{\eta}{}^2 \tag{36}$$

with the derivative evaluated by a second-order one-sided difference expression. The nondimensional force on the body is given by

$$\underset{\sim}{F} = - \oint C_p \, \underset{\sim}{n} \, ds \tag{37}$$

where $\underset{\sim}{n}$ is the unit outward normal to the surface, and ds is an increment of arc length along the surface. The lift and drag coefficients are

$$C_L = \oint C_p \, (-x_\zeta \cos\theta - y_\zeta \sin\theta) d\zeta \tag{38a}$$

$$C_D = \oint C_p \, (y_\zeta \cos\theta + x_\zeta \sin\theta) d\zeta \tag{38b}$$

These integrals were evaluated by numerical quadrature using the trapezoidal rule.

The coordinate system for a Karman-Trefftz airfoil having an integral flap is shown in Fig. 30b, and the streamlines and pressure distribution for this airfoil are compared with the analytic solution (Ref. 20) in Figure 39. Similar excellent comparisons have been obtained with other Karman-Trefftz airfoils. Fig. 32 shows the coordinate system for a Liebeck laminar airfoil, the solution for which is compared with experimental results (Ref. 21) for the pressure distribution in Fig. 40. Finally the coordinate system for a multi-element airfoil is shown in Fig. 35, with the streamlines and pressure distributions shown in Fig. 41. Here coordinate system control was employed as discussed above to attract the coordinate lines into the concave region formed by the intersections of the cut between the airfoils, as well as to the bodies.

Figure 42 shows a coordinate system for a pair of circular cylinders, the coordinate lines being attracted to the intersections of the cut between

the bodies with their surfaces. (The accompanying diagram shows the arrangement of the body and re-entrant segments in the transformed plane.) The streamlines for potential flow obtained on this coordinate system are shown in Figure 43a, and the surface pressure distribution is compared with the analytic solution (Ref. 22) in Figure 43b. By contrast, the uncontracted coordinate system, generated by Eq. (11), and resulting pressure distribution are shown in Figure 44. The effectiveness of the coordinate system control is clear in the comparison of these results with those in Figure 43b. To illustrate the use of different segment arrangements, Figures 45 and 46 show another coordinate system and pressure distribution for the same two cylinders.

The effects of the various numerical parameters involved for the potential flow solution were investigated in some detail, and the results are reported in Ref. 19. These results should serve as a guide to reasonable choices of such parameters as field size, convergence criteria, mesh spacing, etc. for the use of the body-fitted coordinate systems in other applications as well. Ref. 19 also serves to illustrate in some detail the procedure of application of the body-fitted coordinate system to the solution of a partial differential system.

Viscous Flow

The time-dependent, two-dimensional viscous incompressible flow about any number of bodies may be described by the Navier-Stokes equations in various formulations, two of which are illustrated below.

Vorticity-Stream Function Formulation. (Ref. 11 - 14) with the vorticity and stream function as dependent variables the transformed Navier-Stokes equations are

$$\omega_t + (\psi_\eta \omega_\xi - \psi_\xi \omega_\eta)/J = (\alpha\omega_{\xi\xi} - 2\beta\omega_{\xi\eta}$$

$$+ \gamma\omega_{\eta\eta} + \sigma\omega_\eta + \tau\omega_\xi)/J^2 R \tag{39}$$

$$\alpha\psi_{\xi\xi} - 2\beta\psi_{\xi\eta} + \gamma\psi_{\eta\eta} + \sigma\psi_\eta + \tau\psi_\xi = -J^2\omega \tag{40}$$

with boundary conditions:

$$\psi = \text{constant},\ \frac{\sqrt{\gamma}}{J}\psi_\eta = 0 \text{ on body surface} \tag{41a}$$

$$\psi = y\cos\theta - x\sin\theta,\ \omega = 0 \text{ on remote boundary} \tag{41b}$$

All quantities are non-dimensionalized with respect to the free stream velocity and the airfoil chord. All space derivatives in the field were represented by second-order, central difference expressions. The time derivatives were represented by two-point backward difference expressions. The η-derivatives on the body surface were represented by second-order one-sided difference expressions. The solution was implicit in time, all the difference equations being solved simultaneously by SOR iteration at each time step.

The boundary conditions were implemented directly except for the second of (41a), which was satisfied by adjusting the value of the vorticity on the body by a false-position iteration procedure until the second-order, one-sided difference representation of the tangential velocity, $\frac{\sqrt{\gamma}}{J}\psi_\eta$, was below some tolerance:

$$\omega_{i1}^{(k+1)} = \omega_{i1}^{(k)} - K \frac{\omega_{i1}^{(k)} - \omega_{i1}^{(k-1)}}{\left(\frac{\sqrt{\gamma}}{J}\psi_\eta\right)_{i1}^{(k)} - \left(\frac{\sqrt{\gamma}}{J}\psi_\eta\right)_{i1}^{(k-1)}} \left(\frac{\sqrt{\gamma}}{J}\psi_\eta\right)_{i1}^{(k)} \tag{42}$$

Here (k) is the iteration count, K an adjustable parameter, and (i,1) refers to a point on the body surface.

The surface pressure is calculated from the line integral of the Navier-Stokes equations on the surface:

$$p_2 - p_1 = -\frac{2}{R}\int_{\underset{\sim}{r}_1}^{\underset{\sim}{r}_2} (\underset{\sim}{\nabla} \times \underset{\sim}{\omega})\cdot d\underset{\sim}{r}$$

$$= \frac{2}{RJ}\int_{\xi_1}^{\xi_2} (\beta\omega_\xi - \gamma\omega_\eta)d\xi \tag{43}$$

The body force components are then obtained from the integration of the pressure and shear forces around the body surface:

$$F_x = + \oint py_\xi \, d\xi - \frac{2}{R}\oint \omega x_\xi \, d\xi \tag{44a}$$

$$F_y = - \oint px_\xi \, d\xi - \frac{2}{R}\oint \omega y_\xi \, d\xi \tag{44b}$$

Finally, the lift and drag coefficients are given by

$$C_L = F_y\cos\theta - F_x\sin\theta \tag{45a}$$

$$C_D = F_y\sin\theta + F_x\cos\theta \tag{45b}$$

where θ is the angle of attack.

The coordinate system for a Göttingen 625 airfoil shown in Fig. 33 was used in this solution. The high density of constant η-lines near the airfoil is the result of contraction to the first 15 η-lines. Streamline contours are shown in Fig. 47, and velocity profiles are shown in Fig. 48. Pressure and force coefficients are illustrated in Fig. 49.

To show that the boundary-fitted coordinate system can be used with arbitrary shaped bodies, the viscous flow about a cambered rock at a Reynolds number of 500 was developed. The contracted coordinate system used in the solution is given in Fig. 38. ψ and ω contours are shown in

Figure 50, and velocity profiles are shown in Figure 51.

In order that the pressure be single-valued, it is necessary that the value of the stream function on each body of a multiple-body system be such that the line integral of the Navier-Stokes equation on each body vanishes:

$$0 = \oint \underset{\sim}{\nabla} p \cdot d\underset{\sim}{r} = - \oint [\frac{\partial \underset{\sim}{v}}{\partial t} + \underset{\sim}{\nabla}(\frac{|\underset{\sim}{v}|^2}{2})$$

$$- \underset{\sim}{v} \times \underset{\sim}{\omega} + \frac{1}{R} \underset{\sim}{\nabla} \times \underset{\sim}{\omega}] \cdot d\underset{\sim}{r}$$

$$= - \frac{1}{R} \oint (\underset{\sim}{\nabla} \times \underset{\sim}{\omega}) \cdot d\underset{\sim}{r} \qquad (46)$$

Thus in the transformed plane it must be that

$$\oint (\gamma\omega_\eta - \beta\omega_\xi) d\xi = 0 \qquad (47)$$

on each body. Since this requires a double iteration, i.e., for both ω and ψ on each body, it appears that this formulation is not as well suited for two-body calculations as is the primitive variable formulation that follows.

Velocity-Pressure Formulation (Ref. 23, 14). With the velocity and pressure (primitive variables) as the dependent variables the transformed Navier-Stokes equations are

$$u_t + [y_\eta(u^2)_\xi - y_\xi(u^2)_\eta]/J + [x_\xi(uv)_\eta - x_\eta(uv)_\xi]/J$$

$$+ (y_\eta p_\xi - y_\xi p_\eta)/J = (\alpha u_{\xi\xi} - 2\beta u_{\xi\eta} + \gamma u_{\eta\eta}$$

$$+ \sigma u_\eta + \tau u_\xi)/RJ^2 \qquad (48)$$

$$v_t + [y_\eta(uv)_\xi - y_\xi(uv)_\eta]/J + [x_\xi(v^2)_\eta - x_\eta(v^2)_\xi]/J$$

$$+ (x_\xi p_\eta - x_\eta p_\xi)/J = (\alpha v_{\xi\xi} - 2\beta v_{\xi\eta} + \gamma v_{\eta\eta}$$

$$+ \sigma v_\eta + \tau v_\xi)/RJ^2 \qquad (49)$$

$$\alpha p_{\xi\xi} - 2\beta p_{\xi\eta} + \gamma p_{\eta\eta} + \sigma p_{\eta} + \tau p_{\xi} = - (y_{\eta}u_{\xi} - y_{\xi}u_{\eta})^2$$

$$- 2(x_{\xi}u_{\eta} - x_{\eta}u_{\xi})(y_{\eta}v_{\xi} - y_{\xi}v_{\eta})$$

$$- (x_{\xi}v_{\eta} - x_{\eta}v_{\xi})^2 - J^2 D_t \tag{50}$$

where

$$D \equiv (y_{\eta}u_{\xi} - y_{\xi}u_{\eta} + x_{\xi}v_{\eta} - x_{\eta}v_{\xi})/J \tag{51}$$

Equation (50) is the transformed Poisson equation for the pressure, obtained by taking the divergence of the Navier-Stokes equations.

The boundary conditions are

$$u = v = 0 \text{ on body surface} \tag{52a}$$

$$u = \cos\theta,\ v = \sin\theta,\ p = 0 \text{ on remote boundary} \tag{52b}$$

The pressure at each point on the body was adjusted at each iteration by an amount proportional to the velocity divergence evaluated using second-order one-sided differences for the η-derivatives on the body.

Pressure Distribution and Force Coefficients. The surface pressure distribution is calculated in the vorticity-stream function formulation from the line integral of the Navier-Stokes equation around the body surface. In the velocity-pressure formulation the surface pressure, is, of course, obtained directly. In the velocity-pressure formulation it is necessary to calculate the body vorticity before applying (44) from

$$\omega = - \frac{1}{J}(y_{\xi}v_{\eta} + x_{\xi}u_{\eta}) \tag{53}$$

Figure 35 shows the coordinate system for a multiple airfoil consisting of two Karman-Trefftz airfoils, one simulating a separated flap. Coordinate system control was used to attract the coordinate lines strongly to the first ten lines around the bodies and to the intersections of the cut between the bodies with the trailing edge of the ...e body and the leading edge of the aft body. Velocity vectors and pressure distributions for the viscous flow

solution at Reynolds number 1000 are shown in Figure 52.

Loaded Plate (Ref. 24)

Figure 53 shows a comparison between the numerical and analytic solution (Ref. 25) for deflection contours for a simply supported uniformly loaded triangular flat plate. This problem involves the solution of the biharmonic equation by splitting into two Poisson equations. The transformed Poisson equations appear as in Eq. (40) above. Again all computation was done in the rectangular transformed plane.

VI. INSTRUCTIONS FOR USE - COORDINATE SYSTEMS

The coordinate system is generated by the program TOMCAT with the subroutines BNDRY, CORPLOT, LINWT, ERROR, GUESSA, MAXMIN, PARA, RHS, SOR, PLOT, and SYMBOL. Subroutines PLOT and SYMBOL were added for compatibility between the GOULD and CALCOMP plotters. Each facility will probably have to make minor changes for plotting. A complete set of instructions for the input is included in the listing of TOMCAT.

Core must be set to zero at load time.

Files. The program uses two essential files with internal names 10 and 11. File 11 is used to store a partially converged solution so that the iteration can be continued by a subsequent run. This file need not be retained once the solution has converged.

The converged coordinate system is written on file 10. This should be saved to use as input for PROGRAM FATCAT.

Certain files and additional control statements will also be necessary for the operation of the plotter, and these must be added to fit the user's installation. The program is compatible with both the GOULD and the CALCOMP plotters.

Dimensions. The standard program allows a maximum field size of 70 ξ lines and 60 η lines and requires a core size of 131,000 words for the Langley Research Center's CDC 6000 Series Computer System. Error signals and instructions for modification will be given if these limits are exceeded. The three statements requiring modification for larger fields are separated from the rest of the dimension and data statements of the main program for convenience.

Input Parameters. Most of the input is self-explanatory in the instructions given in the listing of TOMCAT. However, a few additional comments may be in order.

Field Size. The parameter IDISK controls the storage of the converged system on the disk file and also signals the restart of a partially converged solution. The format of the storage of the coordinate system on the disk file is given following IDISK in the instructions.

Plotting. Plotting may be by-passed by setting IPLOT to zero and eliminating certain control cards. The selection of the GOULD, CALCOMP, or other plotter is made by the parameter IPLTR. Recall that the user must also add certain site-dependent control statements to the run stream appropriate to the particular plotter installation. The parameters NUMBR and NUMBR1 allow the

plotted field to be confined to a portion of the actual field by restricting the number of curvilinear coordinate lines plotted. Coordinate lines may be skipped in the plot by adjusting ISKIP1 and ISKIP2. The parameters XB1, XB2, YB1, and YB2 also allow the plotted field to be restricted to the portion of the actual field between limits in the cartesian coordinates.

Initial Guess. The parameter IGES controls the initial guess for the iterative solution of the difference equations. Since these equations are nonlinear, convergence can be obtained only from an initial guess within some neighborhood of the solution. The same initial guess will not in general give convergence for all segment arrangements. The type 0 is suitable for single-body and simply-connected systems, however, as well as for multi-body systems having all body segments on the same curvilinear coordinate line. Types 2 and -40 are widely applicable to multiple-body fields, except those having two bodies in single segments on opposite sides of the transformed field, where types 1 or 4 may be effective. Very strong coordinate attraction near a boundary having a sharp convex corner requires an initial guess having sufficiently closely-spaced lines in the region of line attraction, else the iterates may overlap the boundary. In such a situation the exponential weighted average guess (type IGES > 4) should be used. The lines in the guess will be more strongly contracted as IGES is increased. Note that the use of this type of guess requires that the boundary to which the lines are attracted be located on the bottom or left side of the rectangular transformed field. Gradual movement of the outer boundary may also help(see INFAC). See Section III for more information.

Body Contours. Points on the body and outer boundary contours may be placed as desired around the contours, and the cartesian coordinates of these points are input in order from cards, one card per point, or from a file with one image per point.

Some of the contours of a multiple-body system may be split into several

segments which may be located on the rectangular boundary of the transformed field in many different ways as noted above in Section III. For single bodies the body and outer boundary contours are simply cut and opened into single segments. The points will be placed on the rectangular boundary from the first index, LB1, to the second index, LB2, even when LB1 exceeds LB2. The contour segments must be arranged so that the segment ends are connected by coordinate lines that do not cross. This is simply a matter of arranging the segments on the rectangular transformed field boundary such that a continuous path is traversed over the contour segments and connecting cuts in the physical field as a closed circuit is made of the rectangular boundary of the transformed field. The order in which these sets are input is immaterial, except that the outer boundary contour must be last. There is no relation between the order of input of the sets and the order of their appearance on the circuit of the rectangular boundary.

Note that no points are repeated in the input; the closure of each body contour is accomplished internally by the program. (The total number of ξ and η lines, IMAX and JMAX, however, will include the repeated points that close the contours. See, for example , the test cases given following the program listings in this section.

If a circular outer boundary is desired, this contour may be calculated internally rather than being input. In this case the radius and origin of the circle, and the angle of its initial point counter-clockwise with respect to the positive x-axis, are input. The points on the outer boundary contour will then be placed at equal angular increments clockwise from this initial angle. This outer boundary may then be located on the rectangular boundary in one or more segments in the same manner described above.

Re-entrant Boundaries. The re-entrant segments pairs are specified by their end points and the sides on which they lie, but no points are input

thereon, since these are actually cuts rather than boundaries in the physical plane. The order in which the re-entrant segments are input bears no relation either to the order in which these segments occur on the circuit of the rectangular boundary or to the order in which the body contours are input.

Acceleration Paramaters. If a non-zero value is input for R(1), then this value will be used as a uniform SOR acceleration parameter. Typical values for a number of cases have been given above in the Section III. The program also has the capability of calculating a field of variable local acceleration parameters which are updated at each iteration until the maximum absolute change of acceleration parameter over the field is less than the input value R(10), after which the acceleration parameter field is frozen. Since these local parameters are calculated from linear theory, they are not true optimum values. Furthermore, in certain local situations, not even the linear optimum is known. A choice is given, via IEV, for these situations, but under-relaxation is generally the safest course. As noted in Section III, in some cases these calculated variable acceleration parameters tend to be too high and may not give convergence. The use of the variable acceleration parameters also requires extra computer time for their calculation, of course, and this calculation involves a square root. Therefore, the constant input acceleration parameter is usually to be preferred, provided this value is selected with some care with attention to the results in Section III.

Coordinate System Control. The curvilinear coordinate lines may be concentrated by attracting the lines to the body contours or other lines or to points in the field. Generally an effective way for concentration in the

vicinity of a body contour is to use attraction to the contour and also to the first several lines off the contour, with decreasing attraction amplitude on each line outward and a decay factor of 0.5 or less on all lines.

On a field that is 10 chords is radius with 40 lines surrounding the body, an attraction amplitude of 1000 with attraction to 10 lines gives moderate concentration, while 100,000 gives very strong concentration. Amplitudes of 100 or below give only slight changes from the concentration that is inherent in the basic homogeneous equations.

Some attention must be paid to the rapidity of the change of coordinate line spacing with strong attraction else truncation error in the form of artificial diffusion may be introduced as follows: Consider the finite difference approximation of a first derivative with variation only in the x-direction to which the ξ-lines are normal. Then by (8)

$$f_x = \frac{y_\eta f_\xi}{x_\xi y_\eta} = \frac{f_\xi}{x_\xi}$$

The difference approximation then would be

$$f_x = \frac{f_{i+1} - f_{i-1}}{x_{i+1} - x_{i-1}} + T_i$$

where T_i is the local truncation error.

Taylor series expansions of f_{i+1} and f_{i-1} about f_i then yield, after some algebraic rearrangement,

$$T_i = -\frac{1}{2}(f_{xx})_i\,(x_{i+1} + x_{i-1} - 2x_i)$$

But the last factor is simply the difference approximations of $x_{\xi\xi}$ so that

$$T = -\frac{1}{2}x_{\xi\xi}\,f_{xx}$$

This truncation error thus introduces a numerical diffusive effect in the difference approximation of first derivatives. Care must therefore be taken that the second derivatives of the physical coordinates (i.e., the rate of change of the physical spacing between curvilinear coordinate lines) are not too large in regions where the dependent variables have significant second derivatives in the direction normal to the closely spaced coordinate lines.

Just what is a permissible upper limit to the rate of change of the line spacing is problem dependent. Consider, for instance, viscous flow past a finite flat plate parallel to the x-direction. Here the velocity parallel to the wall changes rapidly from zero at the wall to its free stream value over a small distance that is of the order of $\frac{1}{\sqrt{R}}$ where R is the Reynolds number, $R = \frac{U_\infty x}{\nu}$, based on freestream velocity, U_∞ the distance from the leading edge of the plate, x, and the kinematic viscosity, ν.

The equation for the time rate of change of the velocity parallel to the wall is

$$u_t = -uu_x - vu_y + \frac{1}{R}(u_{xx} + u_{yy})$$

Recalling that the large spacial variation in velocity occurs in the y-direction, coordinate lines would be contracted near the plate. The truncation error introduced by this contraction would be

$$-v(-T) = (-\frac{v}{2} y_{\eta\eta})u_{yy}$$

This introduces a negative numerical viscosity $(-\frac{v}{2} y_{\eta\eta})$, since v and $y_{\eta\eta}$ are both positive.

The effective viscosity is thus reduced (effective Reynolds number increased), so that the velocity gradient near the wall is steepened. Therefore care should be taken that $y_{\eta\eta}$ is limited so that the numerical viscosity $(-\frac{v}{2} y_{\eta\eta})$ not significant in comparison with the physical viscosity $(\frac{1}{R})$.

The situation is mitigated somewhat of the fact that the numerical viscosity is proportioned to the small velocity normal to the wall, this velocity being of order $\frac{1}{\sqrt{R}}$. Actually this limit is conservative, since the normal velocity drops to zero at the wall and only attains the order $\frac{1}{\sqrt{R}}$ in the outer portion of the region of large gradient of velocity parallel to the wall where u_{yy} is very small.

Sufficiently close spacing of lines can be obtained even subject to such limits on the rate of change of the spacing by using decay factors in

the tenths range for the coordinate attraction.

The use of coordinate system control tends to slow the convergence of the iterative solution, and it is necessary to add the attraction gradually for strong concentrations. Convergence can be achieved even with very strong attraction amplitude by successively partially converging the field with a weaker amplitude and then using this result as the initial guess for the iteration with a stronger amplitude. This can all be done in one run by inputing the number of steps to be used for addition of the full amplitude (IFAC) and the multiple of the final convergence criterion to be used as the criterion for the partial convergence of each succeeding amplitude (EFAC). Generally the lowest or perhaps the next-to-the-lowest number of steps that will produce convergence is the most economical. In typical single-body fields, an amplitude of 1000 has required three steps, while 10,000 has required six steps. A value of 100.0 is typical for EFAC.

When very strong coordinate attraction is used to a boundary having a sharp convex corner the lines may tend to overlap the corner unless the SOR iterative sweep is toward this boundary. Since the sweep is done toward lower ξ and η values, such a boundary should be located on the bottom or left side of the rectangular transformed plane if strong attraction thereto is to be used. In such a case the initial guess should be IGES > 4 as mentioned above, the stronger the attraction, the larger IGES. When IGES is large enough it should not be necessary to use gradual addition of the attraction. Movement of the outer boundary may also help (INFAC).

Convergence of Very Large Fields. With some segment arrangements for multiple-body fields convergence problems have occured with large fields (20 chords or so). This problem arises since with some arrangements, fewer lines pass between the bodies and the outer boundary in some directions than in others. Therefore, provision has been made for approaching con-

vergence on the final field by successively partially converging smaller fields and using each succeeding result to produce an initial guess by linear projection for the next larger field. This can be done in a single run by specifying INFAC and INFACO. A choice is given between doubling the field radius at each step and increasing it linearly. In the former case the initial size is completely determined by the number of steps specified, while in the latter case it is necessary also to specify the initial point in the linear increase from zero at which the radius is to start. Care should be exercised that the initial outer boundary does not intersect the bodies.

Coordinate System

Program TOMCAT

```
      PROGRAM TOMCAT(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
     1TAPE10,TAPE11)
C   ********* MISSISSIPPI STATE 2-D BODY-FITTED COORDINATE SYSTEM ********
C   *                  (SIMPLY OR MULTIPLY CONNECTED REGIONS)
C   *
C   *       (DEPARTMENT OF AEROPHYSICS AND AEROSPACE ENGINEERING   )
C   *       (         MISSISSIPPI STATE UNIVERSITY      1975         )
C   *       (DEVELOPMENT SPONSORED BY NASA,LANGLEY RESEARCH CENTER)
C   *
C   *            DIRECT INQUIRIES TO DR. JOE F. THOMPSON
C   *                                DRAWER A
C   *                                MISSISSIPPI STATE, MS 39762
C   *
C   *                                PHONE: 601-325-3625
C   *
C   ************************************************************************
C
      DIMENSION X(70,60), Y(70,60), RETA(70,60), RXI(70,60), MACC(70,60)
     1, TACC(70,60), XPLOT(72), YPLOT(72)
      DIMENSION R(13),IXER(2),IYER(2),C1(8),C2(8),BDY(8),          LBSID(6
     1), LB1(6), LB2(6), LBDY(6), LBSID(6), LR1(6), LR2(6), LTSID(6), LI
     21(6), LI2(6), LTYPE(6), LSEN(6), IMER(2)
      INTEGER TACC
C*****
      DATA NDIM,NDIM1 /70,60/
      DATA RAD/57.2957795130B/
      DATA MNBSEG,MNRBEG /6,6/
      DATA ZERO /1.0E-08/
C
C   ************************************************************************
C   *
C   ***************************** INPUT DATA *******************************
C   *
C   *** CARDS(3) .  C1/C2/BDY    - FORMAT(8A10)
C   *                (MAY BE BLANK)
C   *
C   *          C1 AND C2 - 80 CHARACTER A/N ARRAYS WHICH ARE PRINTED AT
C   *                    THE TOP OF EACH OUTPUT PAGE AND ON ANY PLOTS.
C   *
C   *          BDY - NAME OF BODY BEING TRANSFORMED (80 CHARACTERS MAX).
C   *
C   *** CARD : IMAX,JMAX,NBDY,ITER,IGES,IDISK,IWIR,IWINTL,IWFIN,IGED
C   *           - FORMAT(10I5)
C   *
C   *          IMAX - NUMBER OF XI-LINES.
C   *
C   *          JMAX - NUMBER OF ETA-LINES.
C   *
C   *          NBDY - NUMBER OF BODIES IN THE FIELD.
C   *                 (ZERO FOR SIMPLY-CONNECTED REGION)
C   *
C   *          ITER - MAXIMUM NUMBER OF ITERATIONS ALLOWED.
C   *
C   *          IGES - INITIAL GUESS TYPE : (SEE IGED ALSO)
C   *                 (SOME SEGMENT ARRANGEMENTS WILL NOT CONVERGE
C   *                  WITH THE STANDARD INITIAL GUESS, THEREFORE
C   *                  SEVERAL ALTERNATIVES ARE PROVIDED.)
C   *                 (IGES=0 IS TYPICAL FOR SINGLE-BODY FIELDS OR
C   *                  MULTIPLE-BODY FIELDS HAVING ALL BODIES ON THE
C   *                  SAME SIDE OF THE TRANSFORMED FIELD. IGES=2 OR -40
```

```
C  *                  IS TYPICAL FOR OTHER MULTIPLE-BODY FIELDS, EXCEPT
C  *                  FOR THOSE HAVING TWO BODIES IN SINGLE SEGMENTS
C  *                  ON OPPOSITE SIDES OF THE TRANSFORMED FIELD. IN THE
C  *                  LATTER CASE TRY IGES=1 OR 4.)
C  *                 (IGES.NE.0 IS MORE EFFECTIVE FOR CASES WITH THE
C  *                  OUTER BOUNDARY ON THREE SIDES.)
C  *
C  *                 =1 - WEIGHTED AVERAGE OF FOUR PROJECTED
C  *                      BOUNDARY VALUES.
C  *                 =0 - SAME AS 1 EXCEPT ZERO USED IN PLACE OF VALUES
C  *                      ON CUTS.
C  *                 =2 - SAME AS 1 EXCEPT BOUNDARY VALUES ON CUTS
C  *                      OMITTED IN AVERAGE.
C  *                 =3 - MOMENT PROJECTION:
C  *                      X=(SUMD*SUMX-SUMXD)/SUMD, DIVIDED BY TOTAL
C  *                      NUMBER OF BOUNDARY POINTS,
C  *                      WHERE SUMX=SUM OF BOUNDARY VALUES,
C  *                            SUMD=SUM OF DISTANCES TO BOUNDARY POINTS,
C  *                            SUMXD=SUM OF PRODUCTS OF ABOVE.
C  *                 =4 - SAME AS 3 EXCEPT NO DIVISION BY TOTAL
C  *                      NUMBER OF BOUNDARY POINTS.
C  *                 >4 - SAME AS 2 EXCEPT EXPONENTIAL WEIGHT RATHER THAN
C  *                      LINEAR, CONCENTRATION IS TOWARD LOWER VALUES
C  *                      OF XI AND/OR ETA WITH DECAY FACTOR OF
C  *                      0.1*(IGES-4) .
C  *                 <0 - EXPONENTIAL PROJECTION:
C  *                      X=SUMXE/SUME, WITH EXPONENTIAL DECAY
C  *                      FACTOR EQUAL TO IABS(IGES),
C  *                      WHERE  SUME=SUM OF EXP(-DECAY*DISTANCE),
C  *                             SUMXE=SUM OF PRODUCT OF BOUNDARY VALUE
C  *                                   AND ABOVE EXP.
C  *                      (DISTANCE IS NONDIMENSIONALIZED RELATIVE
C  *                       TO DIAGONAL OF RECTANGULAR TRANSFORMED FIELD)
C  *
C  *          IDISK - DISK READ/WRITE CONTROL:
C  *                  =0 START ITERATION FROM INITIAL GUESS,
C  *                     DON'T STORE COORDINATE SYSTEM ON DISK.
C  *                  =1 START ITERATION FROM INITIAL GUESS,
C  *                     STORE COORDINATE SYSTEM ON DISK.
C  *                  =2 CONTINUE ITERATION OF A PARTIALLY
C  *                     CONVERGED SOLUTION READ FROM RESTARTFILE
C  *                     ON DISK, STORE COORDINATE SYSTEM ON DISK.
C  *                  =3 AS =2 EXCEPT DON'T STORE COORDINATE SYSTEM.
C  *
C  *
C  *  NOTE: IDISK OF 1 OR 2 CAUSES THE COORDINATE SYSTEM TO BE WRITTEN
C  *  *     TO DISK IN THE FOLLOWING FORMAT:
C  *  *
C  *  *     WRITE(10,1) C1
C  *  *     WRITE(10,1) C2
C  *  *     WRITE(10,1) IMAX,JMAX,NBSEG,NRSEG,LISEG,NBDY
C  *  *     WRITE(10,1) (LBSID(L),LB1(L),LB2(L),LBDY(L),LBEN(L),
C  *  *    1 L=1,NBSEG)
C  *  *     WRITE(10,1) (LRSID(L),LR1(L),LR2(L),LISID(L),LI1(L),LI2(L),
C  *  *    1 LTYPE(L),L=1,NRSEG)
C  *  *     WRITE(10,1) ((X(I,J),I=1,IMAX),J=1,JMAX),
C  *  *    1             ((Y(I,J),I=1,IMAX),J=1,JMAX)
C  *  *
C  *  *     HERE LISEG IS THE TOTAL NUMBER OF BODY SEGMENTS(EXCLUSIVE
C  *  *     OF OUTER BOUNDARY SEGMENTS) PLUS 1; LBEN(L) IS +1 IF
```

```
C  *  *      LB2(L)>LB1(L) AND IS -1 OTHERWISE(LB1 AND LB2 ARE
C  *  *      INTERCHANGED INTERNALLY AFTER LSEN IS SET IF NECESSARY
C  *  *      SO THAT LB2>LB1); LTYPE IS 1,2,3,4,5,OR 6, RESPECTIVELY,
C  *  *      IF THE TWO SEGMENTS OF A REENTRANT PAIR ARE (1) ONE ON
C  *  *      BOTTOM AND ONE ON TOP, (2) BOTH ON BOTTOM, (3) BOTH ON TOP,
C  *  *      (4) ONE ON LEFT AND ONE ON RIGHT, (5) BOTH ON LEFT, OR
C  *  *      (6) BOTH ON RIGHT; X AND Y ARE THE CARTESIAN COORDINATES.
C  *  *
C  *  *      IF CONVERGENCE IS NOT ATTAINED IN THE ALLOWED NUMBER OF
C  *  *      ITERATIONS THE PARTIALLY CONVERGED SOLUTION IS STORED ON
C  *  *      DISK FOR RESTART, THE ITERATION MAY THEN BE CONTINUED BY
C  *  ***** SETTING IDISK TO 2 OR 3 AND INCREASING ITER.
C  *
C  *          IWIR = =0 DON'T PRINT EACH ITERATION ERROR NORM.
C  *                 =1 PRINT EACH ITERATION ERROR NORM.
C  *
C  *          IWINTL = =0 DON'T PRINT INITIAL GUESS.
C  *                   =1 PRINT INITIAL GUESS.
C  *
C  *          IWPIN = =0 PRINT COORDINATE SYSTEM.
C  *                  =1 DON'T PRINT COORDINATE SYSTEM.
C  *
C  *          IGED = CONTROLS DIRECTION OF WEIGHTED AVERAGES FOR INITIAL
C  *                 GUESS TYPES 0,1,2, AND .GT.4 ;
C  *
C  *                 = 0 = AVERAGE IN BOTH XI AND ETA DIRECTIONS
C  *                       (FOUR POINT AVERAGE)
C  *                 = 1 = AVERAGE IN ONLY ETA DIRECTION
C  *                       (TWO POINT AVERAGE)
C  *                 = 2 = AVERAGE IN ONLY XI DIRECTION
C  *                       (TWO POINT AVERAGE)
C  *
C  *
C  *** CARD 1 IPLOT,IPLTR,NCOPY,LINWT1,LINWT2,NUMBR,NUMBR1,
C  ***   ISKIP1,ISKIP2, = FORMAT(9I5)
C  *
C  *          IPLOT = PLOT OPTIONS;
C  *                  =0 NO PLOTS (INPUT OF REST OF CARD NOT REQUIRED).
C  *                  =1 PLOT COORDINATE SYSTEM.
C  *                  =2 PLOT INITIAL GUESS AND COORDINATE SYSTEM.
C  *
C  *          IPLTR = =1 PLOT WITH GOULD 4800.
C  *                  =2 PLOT WITH GERBER
C  *                  =3 OTHER ( CALL PSEUDO = DEVICE INDEPENDENT =
C  *                         VARIAN AND CALCOMP DO NOT HAVE LINEWT
C  *                         CAPABILITY)
C  *
C  *          NCOPY = NUMBER OF COPIES OF PLOT DESIRED.
C  *
C  *          LINWT1 = PLOT LINE WEIGHT DESIRED FOR PLOT TITLES.
C  *                   (=2 =1, 0, 1, 2 RESPECTIVELY FOR TRIPLE, DOUBLE,
C  *                    NORMAL, HALF, THIRD WEIGHT LINES)
C  *                   (THIS APPLIES ONLY TO THE GOULD 4800)
C  *
C  *          LINWT2 = PLOT LINE WEIGHT DESIRED FOR COORDINATE SYSTEM.
C  *                   (SEE NOTE BELOW LINWT1)
C  *                   (THIS APPLIES ONLY TO THE GOULD 4800)
C  *
C  *          NUMBR = NUMBER OF ETA-LINES DESIRED FOR PLOT.
C  *                  (ZERO VALUE PLOTS ALL)
```

```
C   *
C   *          NUMBR1 = NUMBER OF XI-LINES DESIRED FOR PLOT.
C   *                   (ZERO VALUE PLOTS ALL)
C   *
C   *          ISKIP1 = SKIP PARAMETER FOR XI-LINES.
C   *                   (1 PLOTS EACH LINE, 2 PLOTS EVERY SECOND LINE,ETC.
C   *
C   *          ISKIP2 = SKIP PARAMETER FOR ETA-LINES.
C   *                   (SEE NOTE BELOW ISKIP1)
C   *
C   *** EACH BODY AND THE OUTER BOUNDARY IS DIVIDED INTO ONE OR
C   *** MORE SEGMENTS PLACED ON THE SIDES OF THE RECTANGULAR TRANSFORMED
C   *** FIELD, THESE SEGMENTS ARE CONNECTED BY RE-ENTRANT SEGMENTS.
C   *** THE SEGMENT CONFIGURATION INPUT IS AS FOLLOWS:
C   *
C   *** CARD : NBSEG,NRSEG = FORMAT(2I5)
C   *
C   *          NBSEG = TOTAL NUMBER OF BODY AND OUTER BOUNDARY SEGMENTS.
C   *
C   *          NRSEG = TOTAL NUMBER OF PAIRS OF RE-ENTRANT SEGMENTS.
C   *                  (ZERO FOR SIMPLY-CONNECTED REGION)
C   *
C   *** CARDS(NBSEG) : LBSID,LB1,LB2,LBDY = FORMAT(4I5)
C   *
C   *          LBSID = SIDE OF RECTANGULAR TRANSFORMED FIELD ON WHICH
C   *                  BODY SEGMENT OR OUTER BOUNDARY SEGMENT IS LOCATED.
C   *                  (BOTTOM IS 1, LEFT IS 2, TOP IS 3, RIGHT IS 4.)
C   *
C   *          LB1,LB2 = FIRST AND LAST INDICES OF SEGMENT.
C   *                    (LB1 MAY EXCEED LB2)
C   *
C   *          LBDY = BODY NUMBER (BODIES ARE NUMBERED CONSECUTIVELY
C   *                  FROM  1  TO NBDY
C   *                 OUTER BOUNDARY IS BODY NUMBER 0 OR NBDY+1.
C   *                 NUMBER 0 CAUSES OUTER BOUNDARY TO BE CALCULATED
C   *                 INTERNALLY AS A CIRCLE, POSITIVE OR NEGATIVE
C   *                 NBDY+1 CAUSES OUTER BOUNDARY TO BE READ AS OTHER
C   *                 BODIES.)
C   *                 NOTE : EACH BODY IS READ VIA A SINGLE CARD OR SET
C   *                        OF CARDS WITH NO REPEATED POINTS, NOT EVEN
C   *                        THE FIRST POINT, BODY SEGMENTS MUST BE
C   *                        CONSECUTIVE ACCORDING TO LBDY, WITH THE
C   *                        OUTER BOUNDARY LAST REGARDLESS OF ITS
C   *                        LBDY. POINTS IN EACH SEGMENT ARE READ
C   *                        CONSECUTIVELY AND PLACED ON BOUNDARY
C   *                        FROM INDEX LB1 TO INDEX LB2.
C   *
C   *** CARDS(NRSEG) : LRSID,LR1,LR2,LISID,LI1,LI2 = FORMAT(6I5)
C   *                   (OMIT FOR SIMPLY-CONNECTED REGION)
C   *
C   *          LRSID,LISID = SIDES OF RECTANGULAR TRANSFORMED FIELD ON
C   *                        WHICH SEGMENTS OF A RE-ENTRANT PAIR ARE
C   *                        LOCATED, SEE NOTE BELOW LBSID FOR SIDES.
C   *                        (LISID MUST EQUAL OR EXCEED LRSID)
C   *
C   *          LR1,LR2,LI1,LI2 = FIRST AND LAST INDICES OF EACH SEGMENT
C   *                            OF A RE-ENTRANT PAIR. (IF LISID IS EQUAL
C   *                            TO LRSID THEN LI1 MUST EXCEED LR2, IF
C   *                            LISID IS NOT EQUAL TO LRSID THEN LI1
C   *                            MUST EQUAL LR1. IN ANY CASE LR2 MUST
```

```
C   *                            EXCEED LR1, AND LI2 MUST EXCEED LI1.)
C   *
C   *** CARD : R(1),R(2),R(3),YINFIN,AINFIN,XOINF,YOINF,NINF =
C   *              FORMAT(7F10.0,I5)
C   *
C   *          R(1) = GAUSS-SEIDEL ACCELERATION PARAMETER.
C   *                 ZERO VALUE CAUSES VARIABLE ACCELERATION PARAMETER
C   *                 FIELD TO BE CALCULATED INTERNALLY.
C   *                 (TYPICALLY 1.85)
C   *
C   *          R(2) = CONVERGENCE CRITERION FOR X ITERATION ERROR NORM.
C   *                 (TYPICALLY 0.00001)
C   *
C   *          R(3) = CONVERGENCE CRITERION FOR Y ITERATION ERROR NORM.
C   *                 (TYPICALLY 0.00001)
C   *
C   *          YINFIN = RADIUS OF CIRCULAR OUTER BOUNDARY.
C   *
C   *          AINFIN = ANGLE OF FIRST POINT ON OUTER BOUNDARY. (DEGREES)
C   *                   (ANGLE IS POSITIVE COUNTER-CLOCKWISE FROM POSITIVE
C   *                    X-AXIS. POINTS ARE CLOCKWISE FROM THIS ANGLE.)
C   *
C   *          XOINF,YOINF = CENTER OF CIRCULAR OUTER BOUNDARY.
C   *
C   *          NINF = NUMBER OF UNIQUE POINTS ON OUTER BOUNDARY.
C   *
C   *            NOTE: YINFIN,AINFIN,XOINF, & YOINF MAY BE BLANK IF
C   *                  OUTER BOUNDARY IS TO BE READ.
C   *
C   *** CARD : IEV,IAIT,R(10) = FORMAT(2I5,F10.0)
C   *           (BLANK CARD MAY BE INPUT IF CONSTANT ACCELERATION
C   *            PARAMETER IS USED)
C   *
C   *          IEV = CONTROLS COMPLEX JACOBI EIGENVALUE PROCEDURE.
C   *                (OPTIMUM ACCELERATION WITH REAL EIGENVALUES IS
C   *                 OVER-RELAXATION. OPTIMUM WITH IMAGINARY IS
C   *                 UNDER-RELAXATION. NO THEORETICAL OPTIMUM IS KNOWN
C   *                 FOR COMPLEX EIGENVALUES.)
C   *                 -1 : UNDER-RELAX.
C   *                  0 : WEIGHTED AVERAGE.
C   *                 +1 : OVER-RELAX.
C   *
C   *          IAIT = NON-ZERO VALUE CAUSES VARIABLE ACCELERATION
C   *                 PARAMETER FIELD TO BE PRINTED.
C   *
C   *          R(10) = CONVERGENCE CRITERION FOR VARIABLE ACCELERATION
C   *                  PARAMETER FIELD. FIELD IS FROZEN WHEN MAXIMUM
C   *                  ABSOLUTE CHANGE ON FIELD IS LESS THAN R(10).
C   *                  (TYPICALLY 0.01)
C   *
C   *** CARD : INFAC,INFACO = FORMAT(2I5)
C   *           (CAN BE USED TO AID CONVERGENCE BY CONVERGING A
C   *            SMALLER FIELD FIRST AND USING THIS RESULT TO PRODUCE
C   *            AN INITIAL GUESS FOR A LARGER FIELD. BLANK CARD
C   *            MAY BE INPUT IF THIS FEATURE IS NOT TO BE USED.
C   *            STANDARD IS TO NOT USE THIS FEATURE.)
C   *
C   *          INFAC = NUMBER OF STEPS IN ATTAINMENT OF OUTER BOUNDARY.
C   *                  POSITIVE DOUBLES OUTER BOUNDARY AT EACH STEP.
C   *                  NEGATIVE MOVES OUTER BOUNDARY LINEARLY.
```

```
C  *                  (INCREASE MAGNITUDE IF DIVERGENCE OCCURS)
C  *
C  *          INFACO = INITIAL STEP IN ATTAINMENT OF OUTER BOUNDARY,
C  *                   (LINEAR ATTAINMENT ONLY)
C  *
C  *** CARD : SIZE,RATIO,DIST,XB1,XB2,YB1,YB2 - FORMAT(7F10.0)
C  *          (OMIT IF NO PLOTS DESIRED, I.E., IF IPLOT IS ZERO)
C  *
C  *          SIZE = PLOT SIZE IN Y-DIRECTION, (INCHES)
C  *                 (TYPICALLY 8.0)
C  *
C  *          RATIO = =0 X AND Y PLOT SCALES ARE EQUAL.
C  *                  >0 X AND Y PLOT SCALES ARE ADJUSTED SO THAT THE
C  *                     PLOT IS SQUARE.
C  *                  (TYPICALLY 0)
C  *
C  *          DIST = GOULD PAGES REQUEST(X-DIRECTION), EITHER 10 OR 20.
C  *                 (TYPICALLY 10.0)
C  *
C  *          XB1,XB2 = MINIMUM AND MAXIMUM X-VALUES TO BE PLOTTED.
C  *                    (ZEROS PLOT ALL)
C  *
C  *          YB1,YB2 = MINIMUM AND MAXIMUM Y-VALUES TO BE PLOTTED.
C  *                    (ZEROS PLOT ALL)
C  *
C  *** AFTER THIS INPUT, READ IN BODY COORDINATES - FORMAT(2F10.0)
C  ***
C  *** IF NO COORDINATE SYSTEM CONTROL IS TO BE USED, FOLLOW THESE CARDS
C  *** WITH THREE BLANK CARDS. IF CONTROL IS TO BE USED, USE THE
C  *** FOLLOWING INPUT RATHER THAN THE BLANK CARDS:
C  *
C  *** INPUT FOR COORDINATE SYSTEM CONTROL, USE TWO SETS, ONE FOR
C  *** XI-LINE CONTROL AND ONE FOR ETA-LINE CONTROL.
C  *** (THIS DATA IS READ IN SUBROUTINE SOR.) :
C  *
C  *** CARD : ATYP,ITYP,NLN,NPT,DEC,AMPFAC - FORMAT(A6,I4,2I5,2F10.0)
C  *
C  *          ATYP = TYPE OF ATTRACTION, (XI FOR XI-LINE ATTRACTION,
C  *                 ETA FOR ETA-LINE ATTRACTION,) LEFT JUSTIFIED.
C  *
C  *          ITYP = ZERO GIVES ATTRACTION ON BOTH SIDES.
C  *                 NON-ZERO GIVES ATTRACTION ON CONVEX SIDE AND
C  *                 REPULSION ON CONCAVE SIDE.
C  *
C  *          NLN = NUMBER OF ATTRACTION LINES.
C  *
C  *          NPT = NUMBER OF ATTRACTION POINTS.
C  *
C  *          DEC = NON-ZERO DEC USES DEC FOR DECAY FACTOR.
C  *
C  *          AMPFAC = NON-ZERO AMPFAC MULTIPLIES ALL AMPLITUDES
C  *                   BY AMPFAC.
C  *
C  *** CARDS(NLN) : JLN,ALN,DLN - FORMAT(5X,I5,2F10.0)
C  *                (OMIT IF NLN IS ZERO)
C  *
C  *          JLN = ATTRACTION LINE INDEX.
C  *
C  *          ALN = AMPLITUDE (NEGATIVE REPELS) FOR LINE ATTRACTION.
C  *
```

```
C  *           DLN = DECAY FACTOR FOR LINE ATTRACTION.
C  *
C  *** CARDS(NPT) : IPT,JPT,APT,DPT = FORMAT(2I5,2F10.0)
C  *                  (OMIT IF NPT IS ZERO)
C  *
C  *           IPT,JPT = ATTRACTION POINT INDICES.
C  *
C  *           APT = AMPLITUDE (NEGATIVE REPELS) FOR POINT ATTRACTION.
C  *
C  *           DPT = DECAY FACTOR FOR POINT ATTRACTION.
C  *
C  *** FOLLOW THE COORDINATE SYSTEM CONTROL CARDS WITH THE
C  *** FOLLOWING CARD:
C  *
C  *** CARD : IFAC,IRIT,EFAC = FORMAT(2I5,F10.0)
C  *            (CAN BE USED TO AID CONVERGENCE BY CONVERGING FIELD
C  *             WITH LESS ATTRACTION FIRST AND USING THIS RESULT
C  *             AS THE INITIAL GUESS FOR STRONGER ATTRACTION.
C  *             BLANK CARD MUST BE INPUT IF THIS FEATURE IS NOT USED.
C  *             STANDARD IS TO NOT USE THIS FEATURE , BUT ITS USE MAY
C  *             BE NECESSARY WITH STRONG ATTRACTION.)
C  *
C  *           IFAC = NUMBER OF STEPS IN ADDITION OF INHOMOGENEOUS TERM.
C  *                  POSITIVE DOUBLES INHOMOGENEOUS TERM AT EACH STEP.
C  *                  NEGATIVE INCREASES INHOMOGENEOUS TERM LINEARLY.
C  *                  (INCREASE MAGNITUDE IF DIVERGENCE OCCURS.)
C  *
C  *           IRIT = NON-ZERO VALUE CAUSES INHOMOGENEOUS TERM TO BE
C  *                  PRINTED.
C  *
C  *           EFAC = MULTIPLE OF CONVERGENCE CRITERION TO BE USED FOR
C  *                  INTERMEDIATE CONVERGENCE BETWEEN ADDITIONS OF
C  *                  INHOMOGENEOUS TERM. (TYPICALLY 100.0 )
C  *
C  ******************************************************************
C
C  READ INPUT PARAMETERS
C
      WRITE(6,640)
C
C     A CALL TO PSEUDO INITIALIZES THE GRAPHIC OUTPUT SYSTEM.  THIS MUST
C        BE THE FIRST CALL FOR GRAPHICS IN ORDER TO GENERATE PLOT DATA
C        IN THE FORM OF A DEVICE INDEPENDENT PLOT VECTOR FILE.
C        A CALL TO LEROY SLOWS THE SPEED FOR GERBER FOR LIQUID INK PEN.
C        IT IS OK TO LEAVE CALL LEROY IN FOR THE OTHER DEVICES.
C
      CALL PSEUDO
      CALL LEROY
      READ (5,650) C1,C2,BDY
      READ (5,620) IMAX,JMAX,NBDY,ITER,IGES,IDISK,IWIR,IWINTL,IWFIN,IGED
      IF (IMAX.GT.NDIM) WRITE (6,700) IMAX,NDIM
      IF (JMAX.GT.NDIM1) WRITE (6,720) JMAX,NDIM1
      IF (IMAX.GT.NDIM.OR.JMAX.GT.NDIM1) STOP 1
      READ (5,620) IPLOT,IPLTR,NCOPY,LINWT1,LINWT2,NUMBR,NUMBR1,I
     1SKIP1,ISKIP2
      READ (5,960) NBBEG,NRBEG
      IF (NBBEG.GT.MNBBEG) WRITE (6,910) NBBEG,MNBBEG
      IF (NRBEG.GT.MNRBEG) WRITE (6,920) NRBEG,MNRBEG
      IF (NBBEG.GT.MNBBEG.OR.NRBEG.GT.MNRBEG) STOP 2
      READ (5,880) (LBBID(L),LB1(L),LB2(L),LBDY(L),L=1,NBBEG)
```

```
      IF (NRSEG.NE.0) READ (5,890) (LRSID(L),LR1(L),LR2(L),LISID(L),LI1(
     1L),LI2(L),L=1,NRSEG)
      READ (5,900) (R(I),I=1,3),YINFIN,AINFIN,XOINF,YOINF,NINF
      READ (5,920) IEV,IATT,R(10)
      READ (5,920) INFAC,INFACO
      IF (IPLOT.NE.0) READ(5,930) SIZE,RATIO,DIST,XR1,XR2,YR1,YR2
C
C   WRITE INPUT PARAMETERS
C
      WRITE (6,1060)
      WRITE (6,990) IMAX,JMAX,NBDY,ITER,IGES,IDISK,IMIR,IMINTL,IMFIN
     1,IGEO
      WRITE (6,1000) IPLOT,IPLTM,NCOPY,LINWT1,LINWT2,NUMBR,NUMBR1
     1,ISKIP1,ISKIP2
      WRITE (6,1010) NBSEG,NRSEG
      WRITE (6,1020) R(1),R(2),R(3),YINFIN,AINFIN,XOINF,YOINF,NINF
      WRITE (6,1030) IEV,IATT,R(10)
      WRITE (6,1040) INFAC,INFACO
      WRITE (6,1050) SIZE,RATIO,DIST,XR1,XR2,YR1,YR2
      WRITE (6,1070)
      IF (NRSEG.EQ.0) WRITE (6,1080)
      WRITE (6,970) IGES
      WRITE (6,930) (L,LBSID(L),LB1(L),LB2(L),LBDY(L),L=1,NBSEG)
      IF (NRSEG.NE.0) WRITE (6,940) (L,LRSID(L),LR1(L),LR2(L),LISID(L),L
     1I1(L),LI2(L),L=1,NRSEG)
      IF (INFACO.EQ.0) INFACO=1
      IF (INFAC.EQ.0) INFAC=1
      WRITE (6,950) YINFIN,AINFIN,XOINF,YOINF,NINF,INFAC,INFACO
      IF (R(1).GT.0.0) WRITE (6,870)
C
C   SET UP PARAMETERS
C
      NPLTB=1
      AINFIN=AINFIN/RAD
      IF (NUMBR.EQ.0) NUMBR=JMAX
      IF (NUMBR1.EQ.0) NUMBR1=IMAX
      IF (ISKIP1.EQ.0) ISKIP1=1
      IF (ISKIP2.EQ.0) ISKIP2=1
      IF (NBDY.LE.0) NBDY=1
      IF (IPLOT.EQ.0) GO TO 10
      IF (ABS(XR1).LT.ZERO) XR1=-1.0E20
      IF (ABS(XR2).LT.ZERO) XR2=+1.0E20
      IF (ABS(YR1).LT.ZERO) YR1=-1.0E20
      IF (ABS(YR2).LT.ZERO) YR2=+1.0E20
      IF (DIST.LE.0.0) DIST=10.0
   10 CONTINUE
C
C   READ POINTS ON BODIES AND
C   READ OR CALCULATE POINTS ON OUTER BOUNDARY
C
      DO 20 L=1,NBSEG
   20   IF (IABS(LBDY(L)).EQ.NBDY+1.OR.LBDY(L).EQ.0) GO TO 30
      LTBEG=NBSEG+1
      GO TO 40
   30 LIBEG=L
   40 DO 50 I=1,IMAX
        DO 50 J=1,JMAX
          X(I,J)=0.0
   50     Y(I,J)=0.0
      LBDY=0
```

```
      LBDY(NRBEG+1)=-100
      DO 280 L=1,NRBEG
        I1=LB1(L)
        I2=LB2(L)
        ISEN=1
        IF (I1.GT.I2) ISEN=-1
        LBEN(L)=ISEN
        K1=MIN0(I1,I2)
        K2=MAX0(I1,I2)
        LB1(L)=K1
        LB2(L)=K2
        IF (LBDY(L).EQ.LBDYO) I1=I1+ISEN
        IF (LBDY(L).NE.LBDY(L+1)) I2=I2-ISEN
        IF (LBDY(L).NE.LBDYO) I4=I1
        K1=MIN0(I1,I2)
        K2=MAX0(I1,I2)
        IGOTO=LBSID(L)
        GO TO (60,160,60,160), IGOTO
C**** BOTTOM OR TOP
   60   IF (LBSID(L).EQ.1) J=1
        IF (LBSID(L).EQ.3) J=JMAX
      IF (NRBEG.EQ.0) GO TO 90
      IF (L.LT.LIBEG) GO TO 90
      IF (LBDY(L)) 90,70,90
   70   CALL BNDRY (X,Y,J,I1,I2,LBSID(L),VINFIN,AINFIN,XOINF,YOINF,NINF,
     1NDIM)
        GO TO 130
   90   DO 100 K=K1,K2
          I=I1+(K-K1)*ISEN
  100     READ (5,830) X(I,J),Y(I,J)
  130   IF (LBDY(L).EQ.LBDYO) GO TO 140
        I6=J
        X46=X(I4,I6)
        Y46=Y(I4,I6)
        GO TO 150
  140   CONTINUE
        X(I1-ISEN,J)=X35
        Y(I1-ISEN,J)=Y35
  150   I5=J
        I3=I2
        X35=X(I3,I5)
        Y35=Y(I3,I5)
        IF (LBDY(L).EQ.LBDY(L+1)) GO TO 260
        IF (NRBEG.EQ.0.AND.NBDY.GT.1) GO TO 260
        X(I2+ISEN,J)=X46
        Y(I2+ISEN,J)=Y46
        GO TO 260
C***  LEFT OR RIGHT
  160   IF (LBSID(L).EQ.2) I=1
        IF (LBSID(L).EQ.4) I=IMAX
      IF (NRBEG.EQ.0) GO TO 190
      IF (L.LT.LIBEG) GO TO 190
      IF (LBDY(L)) 190,170,190
  170   CALL BNDRY (X,Y,I,I1,I2,LBSID(L),VINFIN,AINFIN,XOINF,YOINF,NINF,
     1NDIM)
        GO TO 230
  190   DO 200 K=K1,K2
          J=I1+(K-K1)*ISEN
  200     READ (5,830) X(I,J),Y(I,J)
  230   IF (LBDY(L).EQ.LBDYO) GO TO 240
```

```
      I6=I
      X46=X(I6,I4)
      Y46=Y(I6,I4)
      GO TO 250
  240 CONTINUE
      X(I,I1-ISEN)=X35
      Y(I,I1-ISEN)=Y35
  250 I5=I
      I3=I2
      X35=X(I5,I3)
      Y35=Y(I5,I3)
      IF (LBDY(L).EQ.LBDY(L+1)) GO TO 260
      IF (NRSEG.EQ.0.AND.NBDY.GT.1) GO TO 260
      X(I,I2+ISEN)=X46
      Y(I,I2+ISEN)=Y46
  260 IF (L.NE.LISEG-1) GO TO 270
      CALL MAXMIN (X,IMAX,JMAX,NDIM,XBMAX,XBMIN,IDUM,IDUM,IDUM,IDUM,1,
     11)
      CALL MAXMIN (Y,IMAX,JMAX,NDIM,YBMAX,YBMIN,IDUM,IDUM,IDUM,IDUM,1,
     11)
  270 CONTINUE
      LBDYO=LBDY(L)
  280 CONTINUE
     DO 290 L=1,NBSEG
  290 LBDY(L)=IABS(LBDY(L))
C
C  CLASSIFY RE-ENTRANT SEGEMENTS
C
     IF (NRSEG.EQ.0) GO TO 350
     DO 340 L=1,NRSEG
      IGOTO=LRSID(L)
      GO TO (300,310,320,330), IGOTO
  300 LTYPE(L)=1
      IF (LISID(L).EQ.LRSID(L)) LTYPE(L)=2
      GO TO 340
  310 LTYPE(L)=4
      IF (LISID(L).EQ.LRSID(L)) LTYPE(L)=5
      GO TO 340
  320 LTYPE(L)=1
      IF (LISID(L).EQ.LRSID(L)) LTYPE(L)=3
      GO TO 340
  330 LTYPE(L)=4
      IF (LISID(L).EQ.LRSID(L)) LTYPE(L)=6
  340 CONTINUE
  350 CONTINUE
C
C  PULL OUTER BOUNDARY IN FOR INITIAL GUESS IF INFAC NOT ZERO
C
     IF (INFAC.GT.0) CINFAC=1.0/FLOAT(2**(INFAC-1))
     IF (INFAC.LT.0) CINFAC=FLOAT(INFACO)/ABS(FLOAT(INFAC))
     IF (LISEG.GT.NBSEG) GO TO 410
     DO 400 L=LISEG,NBSEG
      I1=LB1(L)
      I2=LB2(L)
      IGOTO=LBSID(L)
      GO TO (360,380,360,380), IGOTO
  360 IF (IGOTO.EQ.1) J=1
      IF (IGOTO.EQ.3) J=JMAX
      DO 370 I=I1,I2
       X(I,J)=X(I,J)*CINFAC
```

```
370       Y(I,J)=Y(I,J)*CINFAC
        GO TO 400
380     IF (IGOTO.EQ.2) I=1
        IF (IGOTO.EQ.4) I=IMAX
        DO 390 J=I1,I2
          X(I,J)=X(I,J)*CINFAC
390       Y(I,J)=Y(I,J)*CINFAC
400     CONTINUE
C
C  INITIAL GUESS
C
  410 CALL GUESSA (IMAX,JMAX,NDIM,X,Y,XBMAX,XBMIN,YBMAX,YBMIN,YINFIN,NBS
     1EG,LB1,LB2,LI1,LI2,LBBID,LIBID,IGES,IGEN)
  405 IF (IPLOT.EQ.2) CALL CORPLT (X,Y,NDIM,NUMBR1,NUMBR,C1,C2,RATIO,SIZ
     1E,NCOPY,LINWT1,LINWT2,ISKIP1,ISKIP2,XPLOT,YPLOT,DIST,IPLTR,NPLTS,
     2 LBBID,LB1,LB2,LBDY,LBBID,LIBID,LB1,LB2,LI1,LI2,LTYPE,NBSEG,N
     3BSEG,LISEG,JMAX,IMAX,XB1,XB2,YB1,YB2)
C
C  PRINT INITIAL DATA
C
  430 WRITE (6,640)
      WRITE (6,660) C1
      WRITE (6,660) C2
      WRITE(6,670) BDY,IMAX,JMAX,R(1),ITER,R(2),R(3)
      WRITE (6,860) NBDY
      IF (NBDY.EQ.1) GO TO 440
  440 IF (IPLOT) 450,450,460
  450 WRITE (6,680)
      GO TO 470
  460 WRITE (6,690) NCOPY,LINWT2,SIZE,RATIO
  470 CONTINUE
      IF (IWINTL.EQ.0) GO TO 500
      WRITE (6,730)
      WRITE (6,740)
      DO 480 J=1,JMAX
        WRITE (6,750) J
  480   WRITE (6,710) (X(I,J),I=1,IMAX)
      WRITE (6,760)
      DO 490 J=1,JMAX
        WRITE (6,750) J
  490   WRITE (6,710) (Y(I,J),I=1,IMAX)
  500 CONTINUE
C
C  ITERATIVE SOLUTION
C
      CALL SOR (R,X,Y,IMAX,JMAX,ITER,IXER,IYER,IEND,NDIM,IWER,NBDY,WAC
     1C,BETA,RXI,LB1,LB2,LI1,LI2,LTYPE,NBSEG,THTR,IAIT,INFAC,INFACD,NBSE
     2G,LISEG,LB1,LB2,LBBID,TACC,IEV,IDISK)
C
C  PRINT FINAL VALUES
C
      IF (IWFIN.NE.0) GO TO 570
      WRITE (6,640)
      WRITE(6,660) C1
      WRITE(6,660) C2
      WRITE (6,770)
      GO TO (510,520,530), IEND
  510 WRITE (6,780)
      GO TO 540
  520 WRITE (6,790) ITER
```

```
      GO TO 540
  530 WRITE (6,800)
  540 WRITE (6,810) R(6),R(7),R(4),R(5),ITER
      WRITE (6,850) IXER(1),IXER(2),IYER(1),IYER(2)
      WRITE (6,740)
      DO 550 J=1,JMAX
        WRITE (6,750) J
  550   WRITE (6,710) (X(I,J),I=1,IMAX)
      WRITE (6,760)
      DO 560 J=1,JMAX
        WRITE (6,750) J
  560   WRITE (6,710) (Y(I,J),I=1,IMAX)
      WRITE (6,640)
  570 IF (IDISK.EQ.0.OR.IDISK.EQ.3) GO TO 600
      WRITE (10,980) C1
      WRITE (10,980) C2
      IF (IDISK.EQ.4) GO TO 580
      WRITE(10,620) IMAX,JMAX,NBSEG,NRSEG,LISEG,NBDY
      WRITE(10,981)(LBBID(L),LB1(L),LB2(L),LBDY(L),LSEN(L),L=1,NBSEG)
      WRITE(10,982)(LRSID(L),LR1(L),LR2(L),LISID(L),LI1(L),LI2(I),
     1LTYPE(L),L=1,NRSEG)
      GO TO 590
  580 WRITE(10,981) IMAX,JMAX
  590 WRITE(10,983)((X(I,J),I=1,IMAX),J=1,JMAX),((Y(I,J),I=1,IMAX),J=1,
     1JMAX)
  600 CONTINUE
C
C  PLOT
C
      IF (IPLOT.GT.0) GO TO 610
      STOP 3
  610 CALL CORPLT (X,Y,NDIM,NUMBR1,NUMBR,C1,C2,RATIO,SIZE,NCOPY,LINWT1,L
     1INWT2,ISKIP1,ISKIP2,XPLOT,YPLOT,DIST,IPLTR,NPLTS,LBBID,LB1,
     2LB2,LBDY,LRSID,LISID,LR1,LR2,LI1,LI2,LTYPE,NBSEG,NRSEG,LISEG,JMAX,
     3IMAX,XB1,XB2,YB1,YB2)
      IF (IPLOT.GT.0) CALL PLOT (0.,0.,999)
      STOP 0101
C
  620 FORMAT (16I5)
  630 FORMAT (8F10.0)
  640 FORMAT (1H1)
  650 FORMAT(8A10)
  660 FORMAT(21X,8A10)
  670 FORMAT (/37X,*BODY-FITTED COORDINATE SYSTEM*//21X,*TRANSFORMED BOD
     1Y, *,8A10//21X,*FIELD PARAMETERS, NUMBER OF XI-LINES =*,I4/38X,* N
     2UMBER OF ETA-LINES = *,I4///19X,*  ITERATION PARAMETERS: SOR ACCEL
     3ERATION PARAMETER = *,F8.5/42X,* MAXIMUM NUMBER OF ITERATIONS ALLO
     4WED = *,I4/39X,*    ALLOWABLE ITERATION ERROR NORMS: X: *,E10.5/75
     5X,* Y: *,E10.5)
  680 FORMAT (/21X,*NO PLOTS DESIRED*)
  690 FORMAT (/21X,*PLOT PARAMETERS:  COPIES DESIRED = *I3/37X,*  LINE*,
     1*WEIGHT DESIRED = *I3/39X,*PLOT SIZE IN Y-DIRECTION = *F5.3/39X,*R
     2ATIO = *,F5.3)
  700 FORMAT (*0*** ERROR **** IMAX TOO LARGE*,10X,*IMAX=*,I5,5X,*MAXIMU
     1M IS*,I5/16X,*INCREASE NDIM IN DATA STATEMENT AND*,* FIRST DIMENSI
     2ON OF X,Y,BETA,RXI,XACC,YACC,*/16X,*ALSO*,* INCREASE DIMENSION OF
     3XPLOT AND YPLOT TO MAXIMUM OF*,* NDIM AND NDIM1 PLUS 2.*)
  710 FORMAT (4X,10F11.5)
  720 FORMAT (*0*** ERROR **** JMAX TOO LARGE*,10X,*JMAX=*,I5,5X,*MAXIMU
     1M IS*,I5/16X,*INCREASE NDIM1 IN DATA STATEMENT AND*,* SECOND DIMEN
```

```
     2SION OF X,Y,BETA,RXI,MACC,TACC,*/16X,*ALSO*,* INCREASE DIMENSION D
     3F XPLOT AND YPLOT TO MAXIMUM OF*,* NDIM AND NDIM1 PLUS 2,*)
 730 FORMAT (/21X,*INITIAL GUESSES FOR X AND Y*)
 740 FORMAT (/5X,33(1H*),9X,*X-ARRAY*8X,53(1H*))
 750 FORMAT (5X,*J=*I4)
 760 FORMAT (/5X,33(1H*),9X,*Y-ARRAY*8X,53(1H*))
 770 FORMAT (/51X,*FINAL VALUES*/)
 780 FORMAT (21X,*ITERATION DIVERGES,*)
 790 FORMAT (21X,*ITERATION IS CONVERGING BUT DOES NOT REACH ERROR TOLE
     1RANCES IN *I3,* ITERATIONS,*)
 800 FORMAT (21X,*ITERATION CONVERGES,*)
 810 FORMAT (/21X,*INITIAL ITERATION ERROR NORMS:  X: *,E10.5,*  Y: *,E
     110.5,* AT ITERATE # 1*/21X,*FINAL ITERATION ERROR*,* NORMS:     X:
     2*,E10.5,*  Y: *,E10.5,* AT ITERATE #*,I4)
 820 FORMAT (2I5,F10.0)
 830 FORMAT (2F10.0)
 850 FORMAT (21X,*LOCATION OF MAXIMUM ITERATION ERROR: X: I=*I5,*,J=*I5
     1/58X,*Y: I=*I5,* J=*I5)
 860 FORMAT (/21X,*NUMBER OF BODIES IN FIELD :*I5)
 870 FORMAT (*0UNIFORM ACCELERATION PARAMETER USED,*)
 880 FORMAT (4I5)
 890 FORMAT (6I5)
 900 FORMAT (7F10.0,2I5)
 910 FORMAT (*0**** ERROR **** NBSEG TOO LARGE*,10X,*NBSEG=*,I3,5X,*MAXI
     1MUM IS*,I3/16X,*INCREASE MNBSEG IN DATA STATEMENT AND*,* DIMENSION
     2S OF LBSID,LB1,LB2,LBDY,LBEN,*)
 920 FORMAT (*0**** ERROR **** NRSEG TOO LARGE*,10X,*NRSEG=*,I3,5X,*MAXI
     1MUM IS*,I3/16X,*INCREASE MNRSEG IN.DATA STATEMENT AND*,* DIMENSION
     2S OF LRSID,LR1,LR2,LISID,LI1,LI2,LTYPE,*)
 930 FORMAT (*0**BODY SEGMENTS***//3X,*L*,2X,*LBSID*,4X,*LB1*,4X,*LB2*,
     13X,*LBDY*//(I4,4I7))
 940 FORMAT (*0**RE-ENTRANT SEGMENTS***//3X,*L*,2X,*LRSID*,4X,*LR1*,4X,
     1*LR2*,2X,*LISID*,4X,*LI1*,4X,*LI2*//(I4,6I7))
 950 FORMAT (*0**OUTER BOUNDARY***//3X,*RADIUS =*,F12.8,10X,*INITIAL AN
     1GLE =*,F13.8/3X,*ORIGIN AT X =*,F11.8,*  , Y =*,F11.8//3X,*NUMBER
     2OF POINTS =*,I4,10X,I5,* STEPS IN*,* ATTAINMENT OF INFINITY*,10X,*
     3INITIAL STEP (LINEAR CASE) =*,I4)
 960 FORMAT (2I5,2F10.0)
 970 FORMAT (*0INITIAL GUESS TYPE:*,I4)
 980 FORMAT(    8A10)
 981 FORMAT(5I5)
 982 FORMAT(7I5)
 983 FORMAT(8E16.8)
 990 FORMAT (*0IMAX,JMAX,NBDY,ITER,IGES,IDISK,IWIR,IWINTL,IWFIN,IGED :*
     1,10I5)
1000 FORMAT (*0IPLOT,IPLTR,NCOPY,LINWT1,LINWT2,NUMBR,NUMBR1,*
     1*ISKIP1,ISKIP2 :*,9I5)
1010 FORMAT (*0NBSEG,NRSEG :*,2I5)
1020 FORMAT (*0R(1),R(2),R(3),YINFIN,AINFIN,XOINF,YOINF,NINF :*/7F15.8,
     1I5)
1030 FORMAT (*0IEV,IAIT,R(10) :*,2I5,F15.8)
1040 FORMAT (*0INPAC,INPACO :*,2I5)
1050 FORMAT (*0SIZE,RATIO,DIST,XB1,XB2,YB1,YB2*,7F12.8)
1060 FORMAT (1X,17(1H*),* INPUT *,30(1H*))
1070 FORMAT (1X,130(1H*))
1080 FORMAT (*0SIMPLY-CONNECTED REGION*)
C
      END
```

```
      SUBROUTINE BNDRY (X,Y,IJ,I1,I2,LSID,R,A,X0,Y0,NINF,N)
C
C  ********************** CIRCULAR OUTER BOUNDARY ************************
C  *
C  * THIS ROUTINE CALCULATES NINF X,Y COORDINATES AROUND A CIRCLE OF
C  * RADIUS R AT EQUALLY SPACED ANGULAR INCREMENTS STARTING AT ANGLE A
C  * (POSITIVE COUNTER-CLOCKWISE FROM POSITIVE X-AXIS) AND PROCEEDING
CLOCLOCKWISE FROM THIS ANGLE.
C  *
C  ***********************************************************************
C
      DIMENSION X(N,1), Y(N,1)
      DATA PI/3.14159265359/
C*****
      K1=MIN0(I1,I2)
      K2=MAX0(I1,I2)
      ISEN=1
      IF (I1.GT.I2) ISEN=-1
      IXM1=NINF
      D=-2.0*PI/FLOAT(IXM1)
      GO TO (10,30,10,30), LSID
C**** BOTTOM OR TOP
   10 DO 20 K=K1,K2
        I=I1+(K-K1)*ISEN
        X(I,IJ)=R*COS(A)+X0
        Y(I,IJ)=R*SIN(A)+Y0
   20   A=A+D
      GO TO 50
   30 DO 40 K=K1,K2
C**** LEFT OR RIGHT
        J=I1+(K-K1)*ISEN
        X(IJ,J)=R*COS(A)+X0
        Y(IJ,J)=R*SIN(A)+Y0
   40   A=A+D
   50 CONTINUE
      RETURN
C
      END




      SUBROUTINE CORPLT (X,Y,NDIM,NUMBR1,NUMBR,C1,C2,RATIO,SIZE,NCOPY,LI
     1NWT1,LINWT2,ISKIP1,ISKIP2,XPLOT,YPLOT,DIST,IPLTR,NPLTS,LBSI
     2D,LB1,LB2,LBDY,LRSID,LISID,LR1,LR2,LI1,LI2,LTYPE,NHSEG,NRSEG,LISEG
     3,JJMAX,IIMAX,XB1,XB2,YB1,YB2)
C
C  *** PLOT ROUTINE - PLOTS COORDINATE SYSTEM AND SEGMENT DIAGRAM ******
C  *
C  *********************************************************************
C
      DIMENSION LBSID(1), LB1(1), LB2(1), LBDY(1), LRSID(1), LISID(1), L
     1R1(1), LR2(1), LI1(1), LI2(1), LTYPE(1)
      DIMENSION X(NDIM,1), Y(NDIM,1), XPLOT(1), YPLOT(1), C1(1), C2(1),
     1AXISL(2), XA(6), YA(6)
      DATA SCAL /0.1/
      DATA S1 /0.0875/
      DATA S2 /0.175/
```

```
      DATA H8 /1.0/
      DATA H9 /0.5/
C*****
      ISY=27
      JMAX=NUMBR
      IMAX=NUMBR1
      H1=0.5*S1
      H2=1.5*S1
      H3=0.5*S2
      H4=1.5*S2
      H5=0.25*S1
      H6=2.0*S1
      H7=1.5*S2
      IMAXM1=IMAX-1
C
C  AXIS MINIMUMS AND SCALE FACTORS
C
      CALL MAXMIN (X,IMAX,JMAX,NDIM,XMAX,XMIN,IXMX,JXMX,IXMN,JXMN,ISKIP1
     1,ISKIP2)
      CALL MAXMIN (Y,IMAX,JMAX,NDIM,YMAX,YMIN,IXMX,JXMX,IXMN,JXMN,ISKIP1
     1,ISKIP2)
      XMAX=AMIN1(XMAX,XR2)
      XMIN=AMAX1(XMIN,XR1)
      YMAX=AMIN1(YMAX,YR2)
      YMIN=AMAX1(YMIN,YR1)
      X1=XMIN
      Y1=YMIN
      AXISL(2)=SIZE
      IF (RATIO) 10,10,20
   10 Y2=(YMAX-YMIN)/AXISL(2)
      X2=Y2
      AXISL(1)=(XMAX-XMIN)/X2
      GO TO 30
   20 AXISL(1)=AXISL(2)
      X2=(XMAX-XMIN)/AXISL(1)
      Y2=(YMAX-YMIN)/AXISL(2)
C
C  SET UP PLOTTER
C
   30 CONTINUE
      NPLTS=2
C
C  LABELS
C
      IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,LINWT1)
      CALL PLOT (.5,.2,-3)
      CALL SYMBOL(0.,.001,.0875,C1,90.,80)
      CALL SYMBOL(.3,.001,.0875,C2,90.,80)
      CALL PLOT (1.,0.,-3)
      CALL PLOT (0.,9.99,2)
      CALL PLOT (.5,.5,-3)
C
C  PLOT LINES OF CONSTANT ETA
C
      IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,LINWT2)
      DO 50 J=1,JMAX,ISKIP2
        K=0
        DO 40 I=1,IMAX,ISKIP1
          IF (X(I,J).GT.XR2.OR.X(I,J).LT.XR1) GO TO 40
          IF (Y(I,J).GT.YR2.OR.Y(I,J).LT.YR1) GO TO 40
```

```
            K=K+1
            XPLOT(K)=X(I,J)
            YPLOT(K)=Y(I,J)
   40     CONTINUE
          XPLOT(K+1)=X1
          XPLOT(K+2)=X2
          YPLOT(K+1)=Y1
          YPLOT(K+2)=Y2
   50 CALL LINE (XPLOT,YPLOT,K,1,0,0,.07)
C
C  PLOT LINES OF CONSTANT XI
C
      DO 70 I=1,IMAX,ISKIP1
        K=0
        DO 60 J=1,JMAX,ISKIP2
          IF (X(I,J).GT.XR2.OR.X(I,J).LT.XR1) GO TO 60
          IF (Y(I,J).GT.YR2.OR.Y(I,J).LT.YR1) GO TO 60
          K=K+1
          XPLOT(K)=X(I,J)
          YPLOT(K)=Y(I,J)
   60     CONTINUE
        XPLOT(K+1)=X1
        XPLOT(K+2)=X2
        YPLOT(K+1)=Y1
        YPLOT(K+2)=Y2
   70 CALL LINE (XPLOT,YPLOT,K,1,0,0,.07)
C
C  SEGMENT CONFIGURATION DIAGRAM
C
      CALL PLOT (AXISL(1)+2.0,1.0,-3)
      JMAX=JJMAX
      IMAX=IIMAX
C
C**** BODY SEGMENTS ****
C
      DO 100 L=1,NBSEG
        D1=LB1(L)*SCAL
        D2=LB2(L)*SCAL
        IGOTO=LBBID(L)
        GO TO (80,90,80,90), IGOTO
   80   IF (LBBID(L).EQ.1) D3=SCAL
C**** BOTTOM OR TOP
        IF (LBBID(L).EQ.3) D3=JMAX*SCAL
        LWT=2
        IF (L.GE.LISEG) LWT=1
      IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,LWT)
        CALL PLOT (D1,D3,3)
        CALL PLOT (D2,D3,2)
      IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,0)
        IF (LBBID(L).EQ.1) H=-H2
        IF (LBBID(L).EQ.3) H=H2+S1
        HH=SIGN(H4,H)+S2*(1.0+SIGN(1.0,H))*0.5
        HHH=SIGN(H5,H)
        CALL NUMBER (D1-H1,D3+H,S1,FLOAT(LB1(L)),0.0,-1)
        CALL NUMBER (D2-H1,D3+H,S1,FLOAT(LB2(L)),0.0,-1)
      IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,-1)
        IF (L.LT.LISEG) CALL NUMBER ((D1+D2)*0.5-H3,D3+HH,S2,FLOAT(LBDY(
     1L)),0.0,-1)
      IF(L.GE.LISEG)CALL SYMBOL((D1+D2)*0.5-H3,D3+HH,S2,IBY,0.0,-1)
      IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,0)
```

```
         CALL PLOT (D1,D3+HHH,3)
         CALL PLOT (D1,D3+HHH,2)
         CALL PLOT (D2,D3+HHH,3)
         CALL PLOT (D2,D3+HHH,2)
         GO TO 100
C**** LEFT OR RIGHT
   90    IF (LBSID(L).EQ.2) D3=SCAL
         IF (LBSID(L).EQ.4) D3=IMAX*SCAL
         LWT=2
         IF (L.GE.LISEG) LWT=1
       IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,LWT)
         CALL PLOT (D3,D1,3)
         CALL PLOT (D3,D2,2)
       IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,0)
         IF (LBSID(L).EQ.2) H=-H6
         IF (LBSID(L).EQ.4) H=H6-S1
         HH=SIGN(H7,H)-S2*(1.0+SIGN(1.0,H))*0.5
         HHH=SIGN(H5,H)
         CALL NUMBER (D3+H,D1,S1,FLOAT(LR1(L)),0.0,-1)
         CALL NUMBER (D3+H,D2,S1,FLOAT(LR2(L)),0.0,-1)
       IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,-1)
         IF (L.LT.LISEG) CALL NUMBER (D3+HH,(D1+D2)*0.5,S2,FLOAT(LRDY(L))
      1,0.0,-1)
       IF(L.GE.LISEG)CALL SYMBOL(D3+HH,(D1+D2)*0.5,S2,ISY,0.0,-1)
       IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,0)
         CALL PLOT (D3+HHH,D1,3)
         CALL PLOT (D3+HHH,D1,2)
         CALL PLOT (D3+HHH,D2,3)
         CALL PLOT (D3+HHH,D2,2)
  100    CONTINUE
C
C**** RE-ENTRANT SEGMENTS ****
C
       IF (NRSEG.EQ.0) GO TO 260
       DO 250 L=1,NRSEG
         D1=LR1(L)*SCAL
         D2=LR2(L)*SCAL
       IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINWT(IPLTR,0)
         IGOTO=LRSID(L)
         GO TO (110,120,110,120), IGOTO
C**** BOTTOM OR TOP - FIRST OF PAIR
  110    IF (LRSID(L).EQ.1) D3=SCAL
         IF (LRSID(L).EQ.3) D3=JMAX*SCAL
         CALL PLOT (D1,D3,3)
         CALL PLOT (D2,D3,2)
         GO TO 130
C**** LEFT OR RIGHT - FIRST OF PAIR
  120    IF (LRSID(L).EQ.2) D3=SCAL
         IF (LRSID(L).EQ.4) D3=IMAX*SCAL
         CALL PLOT (D3,D1,3)
         CALL PLOT (D3,D2,2)
  130    D4=LI1(L)*SCAL
         D5=LI2(L)*SCAL
         IGOTO=LISID(L)
         GO TO (140,150,140,150), IGOTO
C**** BOTTOM OR TOP - SECOND OF PAIR
  140    IF (LISID(L).EQ.1) D6=SCAL
         IF (LISID(L).EQ.3) D6=JMAX*SCAL
         CALL PLOT (D4,D6,3)
         CALL PLOT (D5,D6,2)
```

```
      GO TO 160
C**** LEFT OR RIGHT - SECOND OF PAIR
  150 IF (LTSID(L).EQ.2) D6=SCAL
      IF (LTSID(L).EQ.4) D6=IMAX*SCAL
      CALL PLOT (D6,D4,3)
      CALL PLOT (D6,D5,2)
C
C**** DOTTED LINES CONNECTING RE-ENTRANT SEGMENTS ****
C
  160 IF(IPLTR.EQ.1.OR.IPLTR.EQ.2) CALL LINET(IPLTR,2)
      IGOTO=LTYPE(L)
      GO TO (170,180,190,200,210,220), IGOTO
C**** ONE ON BOTTOM, ONE ON TOP
  170 XA(1)=(D1+D2)*0.5
      YA(1)=D3
      XA(2)=XA(1)
      YA(2)=YA(1)-H8
      XA(3)=IMAX*SCAL+H8
      IF (2.0*D2.LT.FLOAT(IMAX)*SCAL) XA(3)=-H8
      YA(3)=YA(2)
      XA(4)=XA(3)
      YA(4)=JMAX*SCAL+H8
      XA(5)=(D4+D5)*0.5
      YA(5)=YA(4)
      XA(6)=XA(5)
      YA(6)=D6
      N=6
      GO TO 230
C**** BOTH ON BOTTOM
  180 XA(1)=(D1+D2)*0.5
      YA(1)=D3
      XA(2)=XA(1)
      YA(2)=YA(1)-H9
      XA(3)=(D4+D5)*0.5
      YA(3)=YA(2)
      XA(4)=XA(3)
      YA(4)=D6
      N=4
      GO TO 230
C**** BOTH ON TOP
  190 XA(1)=(D1+D2)*0.5
      YA(1)=D3
      XA(2)=XA(1)
      YA(2)=YA(1)+H9
      XA(3)=(D4+D5)*0.5
      YA(3)=YA(2)
      XA(4)=XA(3)
      YA(4)=D6
      N=4
      GO TO 230
C**** ONE ON LEFT, ONE ON RIGHT
  200 XA(1)=D3
      YA(1)=(D1+D2)*0.5
      XA(2)=XA(1)-H8
      YA(2)=YA(1)
      XA(3)=XA(2)
      YA(3)=JMAX*SCAL+H8
      IF (2.0*D2.LT.FLOAT(JMAX)*SCAL) YA(3)=-H8
      XA(4)=IMAX*SCAL+H8
      YA(4)=YA(3)
```

```
          XA(5)=XA(4)
          YA(5)=(D4+D5)*0.5
          XA(6)=D6
          YA(6)=YA(5)
          NA=6
          GO TO 230
C**** BOTH ON LEFT
  210     XA(1)=D3
          YA(1)=(D1+D2)*0.5
          XA(2)=XA(1)-H9
          YA(2)=YA(1)
          XA(3)=XA(2)
          YA(3)=(D4+D5)*0.5
          XA(4)=D6
          YA(4)=YA(3)
          NA=4
          GO TO 230
C**** BOTH ON RIGHT
  220     XA(1)=D3
          YA(1)=(D1+D2)*0.5
          XA(2)=XA(1)+H9
          YA(2)=YA(1)
          XA(3)=XA(2)
          YA(3)=(D4+D5)*0.5
          XA(4)=D6
          YA(4)=YA(3)
          NA=4
C
C  PLOT DOTTED LINES
C
  230     CALL PLOT (XA(1),YA(1),3)
          DO 240 N=2,NA
  240       CALL PLOT (XA(N),YA(N),2)
  250     CONTINUE
C
C  TERMINATION SEQUENCE
C
  260 CALL PLOT (SCAL*FLOAT(IMAX)+2.0,-2.5,-3)
      CALL NFRAME
      RETURN
C
      END




      SUBROUTINE LINWT(IPLTR,IPEN)
C
C  ******************** SETS PLOT LINE WEIGHT **************************
C  *
C  *********************************************************************
C
      GO TO (10,20), IPLTR
   10 CALL LINEWT(IPEN)
      RETURN
   20 CALL PENBL (IPEN)
      RETURN
      END
```

```
      FUNCTION ERROR (VNEW,VOLD,ERROLD,IFLD,JFLD,IVER)
      DIMENSION IVER(1)
C
C   ****************** MAXIMUM NORM OF ITERATE CHANGE *******************
C   *
C   *********************************************************************
C
      T=ABS(VNEW-VOLD)
      IF (T.LE.ERROLD) GO TO 10
      IVER(1)=IFLD
      IVER(2)=JFLD
      ERROR=T
      RETURN
   10 ERROR=ERROLD
      RETURN
C
      END




      SUBROUTINE GUESSA (IMAX,JMAX,NDIM,X,Y,XRMAX,XRMIN,YBMAX,YBMIN,YINF
     1IN,NRSEG,LR1,LR2,LI1,LI2,LRSID,LISID,IGES,IGED)
C
C   ************************** INITIAL GUESS ****************************
C   *
C   *********************************************************************
C
      DIMENSION X(NDIM,1), Y(NDIM,1), LR1(1), LR2(1), LI1(1), LI2(1), LR
     1SID(1), LISID(1)
C*****
      IGESS=IGES
      IF(IGES.EQ.2.OR.IGES.GE.5) IGES=0
      JMXM1=JMAX-1
      IMXM1=IMAX-1
C
C   PROJECTION FROM BODY SEGMENTS
C
      IF (IGES.NE.0) GO TO 30
      T1=0.1*(IGESS-4)
      ARG2=T1*JMXM1
      T2=1.0/(EXP(ARG2)-1.0)
      ARG3=T1*IMXM1
      T3=1.0/(EXP(ARG3)-1.0)
      DFAC=1.0
      DFACC=1.0
      IF(IGED.EQ.1) DFACC=0.0
      IF(IGED.EQ.2) DFAC=0.0
      DO 20 J=2,JMXM1
        IF(IGESS.GE.5) GO TO 6
        FAC=FLOAT(J-1)/FLOAT(JMXM1)
        GO TO 7
    6   ARG1=T1*(J-1)
        FAC=(EXP(ARG1)-1.0)*T2
    7   CONTINUE
        DO 10 I=2,IMXM1
          IF(IGESS.GE.5) GO TO 8
          FACC=FLOAT(I-1)/FLOAT(IMXM1)
```

```
            GO TO 9
    8       ARG1=T1*(I-1)
            FACC=(EXP(ARG1)-1.0)*T3
    9       CONTINUE
            X(I,J)=(X(I,1)+(X(I,JMAX)-X(I,1))*FAC)*DFAC+((X(IMAX,J)-X(1,J)
     1)*FACC+X(1,J))*DFACC
            Y(I,J)=(Y(I,1)+(Y(I,JMAX)-Y(I,1))*FAC)*DFAC+((Y(IMAX,J)-Y(1,J)
     1)*FACC+Y(1,J))*DFACC
            IF (IGESS.EQ.2.OR.IGESS.GE.5) GO TO 10
      IF(DFAC.EQ.0.0.OR.DFACC.EQ.0.0) GO TO 10
            X(I,J)=0.5*X(I,J)
            Y(I,J)=0.5*Y(I,J)
   10       CONTINUE
   20    CONTINUE
C
C  RE-ENTRANT SEGMENTS
C
   30 IF (NRSEG.EQ.0) GO TO 140
        DO 130 L=1,NRSEG
         I1=LR1(L)
         I2=LR2(L)
         II1=I1+1
         II2=I2-1
         IGOTO=LRSID(L)
         GO TO (40,60,40,60), IGOTO
   40    IF (LRSID(L).EQ.1) J=1
         IF (LRSID(L).EQ.3) J=JMAX
         DX=X(I2,J)-X(I1,J)
         DY=Y(I2,J)-Y(I1,J)
         DX=DX/(I2-I1)
         DY=DY/(I2-I1)
         DO 50 I=II1,II2
           DDX=(I-I1)*DX
           DDY=(I-I1)*DY
           X(I,J)=X(I1,J)+DDX
   50      Y(I,J)=Y(I1,J)+DDY
         GO TO 80
   60    IF (LRSID(L).EQ.2) I=1
         IF (LRSID(L).EQ.4) I=IMAX
         DX=X(I,I2)-X(I,I1)
         DY=Y(I,I2)-Y(I,I1)
         DX=DX/(I2-I1)
         DY=DY/(I2-I1)
         DO 70 J=II1,II2
           DDX=(J-I1)*DX
           DDY=(J-I1)*DY
           X(I,J)=X(I,I1)+DDX
   70      Y(I,J)=Y(I,I1)+DDY
   80    I1=LI1(L)
         I2=LI2(L)
         II1=I1+1
         II2=I2-1
         IGOTO=LISID(L)
         GO TO (90,110,90,110), IGOTO
   90    IF (LISID(L).EQ.1) J=1
         IF (LISID(L).EQ.3) J=JMAX
         DX=X(I2,J)-X(I1,J)
         DY=Y(I2,J)-Y(I1,J)
         DX=DX/(I2-I1)
         DY=DY/(I2-I1)
```

```
          DO 100 I=II1,II2
            DDX=(I-I1)*DX
            DDY=(I-I1)*DY
            X(I,J)=X(I1,J)+DDX
  100       Y(I,J)=Y(I1,J)+DDY
          GO TO 130
  110     IF (LIBID(L).EQ.2) I=1
          IF (LIBID(L).EQ.4) I=IMAX
          DX=X(I,I2)-X(I,I1)
          DY=Y(I,I2)-Y(I,I1)
          DX=DX/(I2-I1)
          DY=DY/(I2-I1)
          DO 120 J=II1,II2
            DDX=(J-I1)*DX
            DDY=(J-I1)*DY
            X(I,J)=X(I,I1)+DDX
  120       Y(I,J)=Y(I,I1)+DDY
  130   CONTINUE
  140 IF (IGEO.EQ.0) RETURN
C
C  LINEAR PROJECTION - IGEO=1
C
      IF (IGEO.NE.1) GO TO 170
      DFAC=1.0
      DFACC=1.0
      IF(IGED.EQ.1) DFACC=0.0
      IF(IGED.EQ.2) DFAC=0.0
      DO 160 J=2,JMXM1
        FAC=FLOAT(J-1)/FLOAT(JMXM1)
        DO 150 I=2,IMXM1
          FACC=FLOAT(I-1)/FLOAT(IMXM1)
          X(I,J)=(X(I,1)+(X(I,JMAX)-X(I,1))*FAC)*DFAC+((X(IMAX,J)-X(1,J)
     1)*FACC+X(1,J))*DFACC
          Y(I,J)=(Y(I,1)+(Y(I,JMAX)-Y(I,1))*FAC)*DFAC+((Y(IMAX,J)-Y(1,J)
     1)*FACC+Y(1,J))*DFACC
        IF(DFAC.GT.0.0.OR.DFACC.GT.0.0) GO TO 150
          X(I,J)=0.5*X(I,J)
          Y(I,J)=0.5*Y(I,J)
  150     CONTINUE
  160   CONTINUE
      RETURN
C
C  MOMENT OR EXPONENTIAL PROJECTION
C
  170 BN=2*(IMAX+JMAX)-4
      DM=SQRT(FLOAT((IMAX-1)**2+(JMAX-1)**2))
      DM=(IABS(IGEO))/DM
      DO 210 J=2,JMXM1
        DO 210 I=2,IMXM1
          BX=0.0
          BY=0.0
          BD=0.0
          BXD=0.0
          BYD=0.0
          BXM=0.0
          BYM=0.0
          BE=0.0
          DO 180 II=1,IMAX
            BX=BX+X(II,1)+X(II,JMAX)
            BY=BY+Y(II,1)+Y(II,JMAX)
```

```
              D=IABS(II-I)
              D1=IABS(1-J)
              D2=IABS(JMAX-J)
              D=D**2
              D1=SQRT(D+D1**2)
              D2=SQRT(D+D2**2)
              SD=SD+D1+D2
              SXD=SXD+X(II,1)*D1+X(II,JMAX)*D2
              SYD=SYD+Y(II,1)*D1+Y(II,JMAX)*D2
              E1=EXP(-D1*DM)
              E2=EXP(-D2*DM)
              SXM=SXM+X(II,1)*E1+X(II,JMAX)*E2
              SYM=SYM+Y(II,1)*E1+Y(II,JMAX)*E2
              SE=SE+E1+E2
  180         CONTINUE
            DO 190 JJ=2,JMM1
              SX=SX+X(1,JJ)+X(IMAX,JJ)
              SY=SY+Y(1,JJ)+Y(IMAX,JJ)
              D=IABS(JJ-J)
              D1=IABS(1-I)
              D2=IABS(IMAX-I)
              D=D**2
              D1=SQRT(D+D1**2)
              D2=SQRT(D+D2**2)
              SD=SD+D1+D2
              SXD=SXD+X(1,JJ)*D1+X(IMAX,JJ)*D2
              SYD=SYD+Y(1,JJ)*D1+Y(IMAX,JJ)*D2
              E1=EXP(-D1*DM)
              E2=EXP(-D2*DM)
              SXM=SXM+X(1,JJ)*E1+X(IMAX,JJ)*E2
              SYM=SYM+Y(1,JJ)*E1+Y(IMAX,JJ)*E2
              SE=SE+E1+E2
  190         CONTINUE
            IF (IGEO.EQ.3) GO TO 200
C
C  EXPONENTIAL PROJECTION - IGEO=0
C
            T1=1.0/SE
            X(I,J)=T1*SXM
            Y(I,J)=T1*SYM
         GO TO 210
  200       CONTINUE
C
C  MOMENT PROJECTION - IGEO= 3 OR 4
C
         T1=1.0/SD
         IF (IGEO.EQ.3) T1=T1/SN
            X(I,J)=(SD*SX-SXD)*T1
            Y(I,J)=(SD*SY-SYD)*T1
  210 CONTINUE
      RETURN
C
      END




      SUBROUTINE MAXMIN (X,IMAX,JMAX,NDIM,XMAX,XMIN,IMX,JMX,IMN,JMN,ISKI
     1P1,ISKIP2)
C
```

```
C   **** THIS SUBROUTINE CALCULATES THE MAXIMUM AND MINIMUM VALUES ******
C   **** OF A TWO-DIMENSIONAL DATA ARRAY.                          ******
C   *
C   * INPUT DATA:
C   *
C   * X=2-D DATA ARRAY WHOSE MAXIMUM & MINIMUM IS TO BE DETERMINED
C   * IMAX=LARGEST VALUE OF FIRST SUBSCRIPT OF X TO BE SCANNED
C   * JMAX=LARGEST VALUE OF SECOND   SUBSCRIPT OF X TO BE SCANNED
C   * NDIM=1ST DIMENSION OF X
C   * ISKIP1=SKIP PARAMETER FOR 1ST INDEX OF X (THE I INDEX)
C   * ISKIP2=SKIP PARAMETER FOR 2ND INDEX OF X (THE J INDEX)
C   *
C   * OUTPUT DATA:
C   *
C   * XMAX=MAXIMUM OF X ARRAY
C   * IMX,JMX=I,J LOCATION OF XMAX
C   * XMIN=MINIMUM OF X ARRAY
C   * IMN,JMN=I,J LOCATION OF XMIN
C   *
C   ******************************.*************************************
C
      DIMENSION X(NDIM,1)
C*****
      XMAX=X(1,1)
      XMIN=X(1,1)
      IMX=1
      JMX=1
      IMN=1
      JMN=1
      DO 20 I=1,IMAX,ISKIP1
        DO 20 J=1,JMAX,ISKIP2
          IF (X(I,J).LT.XMAX) GO TO 10
          XMAX=X(I,J)
          IMX=I
          JMX=J
   10     IF (X(I,J).GT.XMIN) GO TO 20
          XMIN=X(I,J)
          IMN=I
          JMN=J
   20     CONTINUE
      RETURN
C
      END




      SUBROUTINE PARA (XXI,YXI,XETA,YETA,LACC,CPAC,RETA,RXI,CSI,CSJ,MACC
     1,R,T,ACCL,ACCL1,NDIM,I,J,IMER,TACC,IEV)
C
C   **** COEFFICIENTS AND VARIABLE ACCELERATION PARAMETERS ON   **********
C   **** RE-ENTRANT SEGMENTS                                    **********
C   *
C   ********************************************************************
C
      DIMENSION RETA(NDIM,1), RXI(NDIM,1), MACC(NDIM,1), R(1), T(1)
      DIMENSION TACC(NDIM,1)
      DIMENSION IMER(1)
      REAL JACOB,JBAG
      INTEGER TACC
```

```
      LOGICAL LACC
C*****
      UACC(RJ)=2.0/(1.0+SQRT(1.0+RJ**2))
      DACC(RJ)=2.0/(1.0+SQRT(1.0-RJ**2))
      ALFA=XETA**2+YETA**2
      GAMA=XXI**2+YXI**2
      AG=1.0/(ALFA+GAMA)
      JACBB=(XXI*YETA-XETA*YXI)**2*CFAC
      JBAG=JACBB*AG*0.0625
C
C  VARIABLE ACCELERATION PARAMETER
C
      IF (LACC) GO TO 110
      TEM1=JACBB*0.125
      B1=-TEM1*RXI(I,J)
      B2=-TEM1*RETA(I,J)
      B1=ABS(B1)
      B2=ABS(B2)
      T1=ALFA**2-B1**2
      T2=GAMA**2-B2**2
      AT1=ABS(T1)
      AT2=ABS(T2)
      IF (T1.GE.0.0.AND.T2.GE.0.0) GO TO 80
      IF (T1.LT.0.0.AND.T2.LT.0.0) GO TO 90
      RJ1=SQRT(AT1)*CSI*AG
      IF (T1.GE.0.0) GO TO 10
      TEMPW1=UACC(RJ1)
      TACC(I,J)=10
      GO TO 20
   10 TEMPW1=DACC(RJ1)
      TACC(I,J)=20
   20 RJ2=SQRT(AT2)*CSJ*AG
      IF (T2.GE.0.0) GO TO 30
      TEMPW2=UACC(RJ2)
      TACC(I,J)=TACC(I,J)+1
      GO TO 40
   30 TEMPW2=DACC(RJ2)
      TACC(I,J)=TACC(I,J)+2
   40 IF (IEV) 50,60,70
   50 TEMPW=UACC(SQRT(RJ1**2+RJ2**2))
      GO TO 100
   60 TEMPW=(RJ1*TEMPW1+RJ2*TEMPW2)/(RJ1+RJ2)
      GO TO 100
   70 TEMPW=DACC(SQRT(RJ1**2+RJ2**2))
      GO TO 100
   80 RJ=(SQRT(AT1)*CSI+SQRT(AT2)*CSJ)*AG
      TEMPW=DACC(RJ)
      TACC(I,J)=2
      GO TO 100
   90 RJ=(SQRT(AT1)*CSI+SQRT(AT2)*CSJ)*AG
      TEMPW=UACC(RJ)
      TACC(I,J)=1
  100 R(11)=ERROR(TEMPW,WACC(I,J),R(11),I,J,IWER)
      WACC(I,J)=TEMPW
  110 CONTINUE
C
C  COEFFICIENTS
C
      T(5)=JBAG*RETA(I,J)
      T(6)=JBAG*RXI(I,J)
```

```
      T(1)=.5*ALFA*AG
      T(2)=.5*GAMA*AG
      T(3)=.25*(XXI*XETA+YXI*YETA)*AG
      ACCL=ACC(I,J)
      ACCL1=1.0-ACCL
      R(12)=R(12)*ACCL
      RETURN
C
      END




      SUBROUTINE RHS (R,IMAX,JMAX,NDIM,NLN,NPT,ATYP,ITYP,DEC,AMPFAC,RETA
     1)
C
C  ******** INHOMOGENEOUS TERM FOR COORDINATE ATTRACTION ***************
C  *
C  *********************************************************************
C
      DIMENSION R(1), RETA(NDIM,1), ALN(20), DLN(20), JLN(20), APT(100),
     1 DPT(100), IPT(100), JPT(100)
      INTEGER XI,ETA,BLK,ATYP
      DATA ZERO/1.0E-10/
      DATA XI,ETA,BLK/6HXI    ,6HETA   ,6H
C*****
      IF (NLN+NPT.EQ.0) RETURN
      WRITE (6,200)
      IF (ITYP.EQ.0) WRITE (6,260)
      IF (ITYP.NE.0) WRITE (6,270)
      IF (ATYP.EQ.ETA) WRITE (6,280)
      IF (ATYP.EQ.XI) WRITE (6,290)
C
C  SET UP AMPLITUDE AND DECAY FACTOR
C
      IF (NLN.NE.0) READ (5,210) (JLN(L),ALN(L),DLN(L),L=1,NLN)
      IF (NPT.NE.0) READ (5,220) (IPT(L),JPT(L),APT(L),DPT(L),L=1,NPT)
      IF (DEC.LT.ZERO) GO TO 40
      IF (NLN.EQ.0) GO TO 20
      DO 10 L=1,NLN
   10   DLN(L)=DEC
   20 IF (NPT.EQ.0) GO TO 40
      DO 30 L=1,NPT
   30   DPT(L)=DEC
   40 CONTINUE
      IF (ABS(AMPFAC).LT.ZERO) GO TO 80
      IF (NLN.EQ.0) GO TO 60
      DO 50 L=1,NLN
   50   ALN(L)=ALN(L)*AMPFAC
   60 IF (NPT.EQ.0) GO TO 80
      DO 70 L=1,NPT
   70   APT(L)=APT(L)*AMPFAC
   80 CONTINUE
      IF (NLN.NE.0) WRITE (6,230) (JLN(L),ALN(L),DLN(L),L=1,NLN)
      IF (NPT.NE.0) WRITE (6,240) (IPT(L),JPT(L),APT(L),DPT(L),L=1,NPT)
C
C  CALCULATE INHOMOGENEOUS TERM
C
      DO 190 I=1,IMAX
        DO 190 J=1,JMAX
```

```
C**** LINE ATTRACTION
            IF (ATYP.EQ.ETA) IJ=J
            IF (ATYP.EQ.XI) IJ=I
            T2=0.0
            IF (NLN.EQ.0) GO TO 100
            DO 90 L=1,NLN
              T=ALN(L)*EXP(-DLN(L)*IABS(IJ-JLN(L)))
              IF (ITYP.NE.0) GO TO 90
              T=T*ISIGN(1,IJ-JLN(L))
              IF (IJ.EQ.JLN(L)) T=0.0
   90         T2=T2+T
C**** POINT ATTRACTION
  100       IF (NPT.EQ.0) GO TO 180
            DO 170 L=1,NPT
              I1=IPT(L)
                IF (ATYP.NE.ETA) GO TO 140
                I1=J-JPT(L)
                GO TO 150
  140           I1=I-IPT(L)
  150           ISN=ISIGN(1,I1)
                IF (I1.EQ.0) ISN=0
                T=(I-IPT(L))**2+(J-JPT(L))**2
                T=APT(L)*EXP(-DPT(L)*SQRT(T))
                IF (ITYP.NE.0) GO TO 160
                T=T*ISN
                IF (I.EQ.IPT(L).AND.J.EQ.JPT(L)) T=0.0
  160           T2=T2+T
  170         CONTINUE
  180       RETA(I,J)=T2
  190       CONTINUE
C
C  PRINT INHOMOGENEOUS TERM
C
      IF (IRIT.NE.0) WRITE (6,250) ((I,J,RETA(I,J),I=1,IMAX),J=1,JMAX)
C*****
      RETURN
C
  200 FORMAT ('0EXPONENTIAL DECAY RHS')
  210 FORMAT (I10,2F10.0)
  220 FORMAT (2I5,2F10.0)
  230 FORMAT ('0ATTRACTION LINES'//4X,'J',17X,'AMP',15X,'DECAY'/(I5,2F20
     1.8))
  240 FORMAT ('0ATTRACTION POINTS'//4X,'I',4X,'J',17X,'AMP',15X,'DECAY',
     1/(2I5,2F20.8))
  250 FORMAT ('0RHS'//6(3X,'I',3X,'J',6X,'RETA **')/(6(2I4,1E10.4,' **')))
  260 FORMAT (' - ATTRACTION **')
  270 FORMAT (' - ATTRACTION TO CONVEX SIDE, REPULSION TO CONCAVE **')
  280 FORMAT ('0*** ETA EQUATION RHS ****')
  290 FORMAT ('0*** XI EQUATION RHS ****')
C
      END




      SUBROUTINE SOR (R,X,Y,IMAX,JMAX,ITER,IXER,IYER,IEND,NDIM,SWER,NB
     1DY,MACC,RETA,RXI,LR1,LR2,LI1,LI2,LTYPE,NRBEG,IWIR,IAIY,INFAC,INFAC
     2D,NBBEG,LIBEG,LB1,LB2,LABID,TACC,TEV,TDISK)
C
C  ********************** SOR ITERATIVE SOLUTION **************************
```

```
C  *
C  *
C  * LOGICAL CONTROLS : LACC - VARIABLE ACCELERATION PARAMETER FIELD
C  *                           CONVERGED, (IRRELEVANT IF LCON=TRUE)
C  *
C  *                    LCFAC - ADDITION OF INHOMOGENEOUS TERM COMPLETED
C  *
C  *                    LCON - CONSTANT ACCELERATION PARAMETER.
C  *
C  * ACCELERATION PARAMETER TYPE : 1 - BOTH EV IMAGINARY (UNDER-RELAX)
C  *                               2 - BOTH EV REAL (OVER-RELAX)
C  *                              12 - XI EV IMAGINARY, ETA EV REAL
C  *                              21 - XI EV REAL, ETA EV IMAGINARY
C  *
C  *****************************************************************
C
      DIMENSION X(NDIM,1), Y(NDIM,1), W(1), IXER(1), IYER(1), T(5), LP1(
     11), LP2(1), LI1(1), LI2(1), LTYPE(1), IWER(1)
      DIMENSION WACC(NDIM,1), RETA(NDIM,1), RXI(NDIM,1), TACC(NDIM,1)
      DIMENSION LB1(1), LB2(1), LBSID(1)
      DOUBLE PRECISION AG
      INTEGER ATYP,ETA,XI,TACC
      REAL JSAG,JACBS
      LOGICAL LACC,LCFAC,LCON
      DATA XI,ETA/6HXI    ,6HETA   /
      DATA PI/3.14159265359/
C*****
      HACK(X,T,I,J,I1,I2,J1,J2,A,A1,T2,T1)=A1*X(I,J)+A*(T(1)*(X(I2,J)+X(
     1I1,J))+T(2)*(X(I,J2)+X(I,J1))-T(3)*(X(I2,J2)-X(I2,J1)+X(I1,J1)-X(I
     21,J2))+T(5)*T2+T(4)*T1)
      HACK2(X,T,I,J,I1,I2,J1,J2,A,A1,T2,T1)=A1*X(I,J)+A*(T(1)*(X(I2,J)+X
     1(I1,J))+T(2)*(X(I,J2)+X(J1,J2))-T(3)*(X(I2,J2)-X(J1-1,J2)+X(J1+1,J
     22)-X(I1,J2))+T(5)*T2+T(4)*T1)
      HACK3(X,T,I,J,I1,I2,J1,J2,A,A1,T2,T1)=A1*X(I,J)+A*(T(1)*(X(I2,J)+X
     1(I1,J))+T(2)*(X(J1,J2)+X(I,J2))-T(3)*(X(J1-1,J2)-X(I2,J2)+X(I1,J2)
     2-X(J1+1,J2))+T(5)*T2+T(4)*T1)
      HACK4(X,T,I,J,I1,I2,J1,J2,A,A1,T2,T1)=A1*X(I,J)+A*(T(1)*(X(I2,J)+X
     1(I2,I1))+T(2)*(X(I,J2)+X(I,J1))-T(3)*(X(I2,J2)-X(I2,J1)+X(I2,I1+1)
     2-X(I2,I1-1))+T(5)*T2+T(4)*T1)
      HACK5(X,T,I,J,I1,I2,J1,J2,A,A1,T2,T1)=A1*X(I,J)+A*(T(1)*(X(I2,I1)+
     1X(I2,J))+T(2)*(X(I,J2)+X(I,J1))-T(3)*(X(I2,I1-1)-X(I2,I1+1)+X(I2,J
     21)-X(I2,J2))+T(5)*T2+T(4)*T1)
      UACC(RJ)=2.0/(1.0+SQRT(1.0+RJ**2))
      OACC(RJ)=2.0/(1.0+SQRT(1.0-RJ**2))
C*****
      WRITE (6,630)
C
C  READ COORDINATE ATTRACTION PARAMETERS AND CALCULATE
C  INHOMOGENEOUS TERM
C
      KO=1
      READ (5,610) ATYP,ITYP,NLN,NPT,DEC,AMPFAC
      IF (ATYP.EQ.ETA) CALL RHS (R,IMAX,JMAX,NDIM,NLN,NPT,ATYP,ITYP,DEC,
     1AMPFAC,RETA)
      IF (ATYP.EQ.XI) CALL RHS (R,IMAX,JMAX,NDIM,NLN,NPT,ATYP,ITYP,DEC,A
     1MPFAC,RXI)
      READ (5,610) ATYP,ITYP,NLN,NPT,DEC,AMPFAC
      IF (ATYP.EQ.ETA) CALL RHS (R,IMAX,JMAX,NDIM,NLN,NPT,ATYP,ITYP,DEC,
     1AMPFAC,RETA)
      IF (ATYP.EQ.XI) CALL RHS (R,IMAX,JMAX,NDIM,NLN,NPT,ATYP,ITYP,DEC,A
```

```
     1MPFAC,RXI)
      READ (5,590) IFAC,IRIT,EFAC
C
C  INITIAL SETUP
C
      IF(R(1).GT.0.0) GO TO 40
      IF (IEV) 10,20,30
   10 WRITE (6,640)
      GO TO 40
   20 WRITE (6,650)
      GO TO 40
   30 WRITE (6,660)
   40 CONTINUE
      IF (IDISK.EQ.2.OR.IDISK.EQ.3) GO TO 50
      GO TO 60
   50 READ (11,720) LACC,LCON,CFAC,LCFAC,IRCOUN,INCOUN,CINFAC,K,IEV,IFAC
     1,EFAC,JMXM1,IMXM1,JMXP1,CSI,CSJ,NPT,INFAC,EINF,IEND
      READ  (11,730) (R(I),I=1,13)
      READ  (11,730)((X(I,J),Y(I,J),RXI(I,J),RETA(I,J),WACC(I,J),I=1,IMA
     1X),J=1,JMAX)
      REWIND 11
      KO=K+1
      IF (IEND.NE.3) GO TO 100
      KO=1
      GO TO 80
   60 IF (IFAC.EQ.0) IFAC=1
      WRITE (6,600) IFAC,EFAC
      EFAC=AMIN1(R(2),R(3))*EFAC
      IF (IFAC.GT.0) CFAC=1.0/FLOAT(2**(IFAC-1))
      IF (IFAC.LT.0) CFAC=1.0/ABS(FLOAT(IFAC))
      WRITE (6,670) CFAC
      IRCOUN=1
      IF (IWR.GT.0) WRITE (6,570)
      JMXM1=JMAX-1
      IMXM1=IMAX-1
      JMXP1=JMAX+1
      CSJ=COS(PI/FLOAT(JMXM1))
      CSI=COS(PI/FLOAT(IMXM1))
      NPT=(IMAX-2)*(JMAX-2)
      IF (NRBEG.EQ.0) GO TO 75
      DO 70 L=1,NRBEG
   70   NPT=NPT+LR2(L)-LR1(L)-1
   75 IF (INFAC.EQ.0) INFAC=1
      IF (INFAC.GT.0) CINFAC=1.0/FLOAT(2**(INFAC-1))
      IF (INFAC.LT.0) CINFAC=FLOAT(INFACO)/ABS(FLOAT(INFAC))
      WRITE (6,710) CINFAC
      INCOUN=1
      IF (INFAC.LT.0) INCOUN=INFACO
      EINF=EFAC
C
C  SET UP ITERATION
C
   80 LCON=R(1).GT.0.0
      LCFAC=.FALSE.
      IF (IABS(IFAC).LE.1) LCFAC=.TRUE.
      DO 90 I=1,IMAX
        DO 90 J=1,JMAX
   90     WACC(I,J)=1.0
      WRITE (6,620)
C
```

```
C   SOR ITERATION
C
  100 IEND=0
      DO 450 K=KO,ITER
        IF (K.LE.2) LACC=.TRUE.
        IF (K.NE.3) GO TO 120
        LACC=.FALSE.
        IF (LCON) LACC=.TRUE.
        TEM=R(1)
        DO 110 I=1,IMAX
          DO 110 J=1,JMAX
  110       MACC(I,J)=TEM
  120   R(4)=0.0
        R(5)=0.0
        R(11)=0.0
        R(12)=0.0
C
C**** FIELD ****
C
        DO 260 JJ=2,JMXM1
        J=JMXM1-JJ+2
          JM1=J-1
          JP1=J+1
          DO 260 III=2,IMXM1
          I=IMXM1-III+2
            IP1=I+1
            IF (I.GT.1) GO TO 130
            IM1=IMXM1
            GO TO 140
  130       IM1=I-1
  140       XXI=X(IP1,J)-X(IM1,J)
            YXI=Y(IP1,J)-Y(IM1,J)
            XETA=X(I,JP1)-X(I,JM1)
            YETA=Y(I,JP1)-Y(I,JM1)
            ALFA=XETA**2+YETA**2
            GAMA=XXI**2+YXI**2
            AG=1.0/(ALFA+GAMA)
            JACBB=(XXI*YETA-XETA*YXI)**2*CFAC
            JBAG=JACBB*AG*0.0625
C**** VARIABLE ACCELERATION PARAMETER
            IF (LACC) GO TO 250
            TEM1=JACBB*0.125
            B1=-TEM1*RXI(I,J)
            B2=-TEM1*RETA(I,J)
            B1=ABS(B1)
            B2=ABS(B2)
            T1=ALFA**2-B1**2
            T2=GAMA**2-B2**2
            AT1=ABS(T1)
            AT2=ABS(T2)
            IF (T1.GE.0.0.AND.T2.GE.0.0) GO TO 220
            IF (T1.LT.0.0.AND.T2.LT.0.0) GO TO 230
            RJ1=SQRT(AT1)*CSI*AG
            IF (T1.GE.0.0) GO TO 150
            TEMPH1=UACC(RJ1)
            TACC(I,J)=10
            GO TO 160
  150       TEMPH1=OACC(RJ1)
            TACC(I,J)=20
  160       RJ2=SQRT(AT2)*CSJ*AG
```

```
      IF (T2.GE.0.0) GO TO 170
      TEMPW2=UACC(RJ2)
      TACC(I,J)=TACC(I,J)+1
      GO TO 180
170   TEMPW2=DACC(RJ2)
      TACC(I,J)=TACC(I,J)+2
180   IF (IEV) 190,200,210
190   TEMPW=UACC(SQRT(RJ1**2+RJ2**2))
      GO TO 240
200   TEMPW=(RJ1*TEMPW1+RJ2*TEMPW2)/(RJ1+RJ2)
      GO TO 240
210   TEMPW=DACC(SQRT(RJ1**2+RJ2**2))
      GO TO 240
220   RJ=(SQRT(AT1)*CSI+SQRT(AT2)*CSJ)*AG
      TEMPW=DACC(RJ)
      TACC(I,J)=2
      GO TO 240
230   RJ=(SQRT(AT1)*CSI+SQRT(AT2)*CSJ)*AG
      TEMPW=UACC(RJ)
      TACC(I,J)=1
240   R(11)=ERROR(TEMPW,WACC(I,J),R(11),I,J,IWER)
      WACC(I,J)=TEMPW
250   CONTINUE
C**** ITERATE
      T(5)=JBAG*RETA(I,J)
      T(4)=JBAG*RXI(I,J)
      T(1)=.5*ALFA*AG
      T(2)=.5*GAMA*AG
      T(3)=.25*(XXI*XETA+YXI*YETA)*AG
      ACCL=WACC(I,J)
      ACCL1=1.0-ACCL
      R(12)=R(12)+ACCL
      TEMPX=HACK(X,T,I,J,IM1,IP1,JM1,JP1,ACCL,ACCL1,XETA,XXI)
      TEMPY=HACK(Y,T,I,J,IM1,IP1,JM1,JP1,ACCL,ACCL1,YETA,YXI)
      R(4)=ERROR(TEMPX,X(I,J),R(4),I,J,IXER)
      R(5)=ERROR(TEMPY,Y(I,J),R(5),I,J,IYER)
      X(I,J)=TEMPX
      Y(I,J)=TEMPY
      IF (I.GT.1) GO TO 260
      X(IMAX,J)=TEMPX
      Y(IMAX,J)=TEMPY
260   CONTINUE
C
C**** REENTRANT SEGMENTS ****
C
    IF (NRSEG.EQ.0) GO TO 400
    DO 390 L=1,NRSEG
      I1=LR1(L)+1
      I2=LR2(L)-1
      I3=LI1(L)+1
      I4=LI2(L)-1
      IGOTO=LTYPE(L)
      GO TO (270,290,310,330,350,370), IGOTO
C**** ONE ON BOTTOM, ONE ON TOP
270   DO 280 JJ=I1,I2
      I=I2-JJ+I1
        XXI=X(I+1,1)-X(I-1,1)
        YXI=Y(I+1,1)-Y(I-1,1)
        XETA=X(I,2)-X(I,JMXM1)
        YETA=Y(I,2)-Y(I,JMXM1)
```

```
            CALL PARA (XXI,YXI,XETA,YETA,LACC,CFAC,RETA,RXI,CSI,CSJ,MACC
     1,R,T,ACCL,ACCL1,NDIM,I,1,IMER,TACC,IEV)
            TEMPX=HACK(X,T,I,1,I-1,I+1,JMXM1,2,ACCL,ACCL1,XETA,XXI)
            TEMPY=HACK(Y,T,I,1,I-1,I+1,JMXM1,2,ACCL,ACCL1,YETA,YXI)
            R(4)=ERROR(TEMPX,X(I,1),R(4),I,1,IXER)
            R(5)=ERROR(TEMPY,Y(I,1),R(5),I,1,IYER)
            X(I,1)=TEMPX
            Y(I,1)=TEMPY
            X(I,JMAX)=TEMPX
            Y(I,JMAX)=TEMPY
  280       CONTINUE
          GO TO 390
C**** BOTH ON BOTTOM
  290     DO 300 JJ=I1,I2
          I=I2-JJ+I1
            II=I4-(I-I1)
            XXI=X(I+1,1)-X(I-1,1)
            YXI=Y(I+1,1)-Y(I-1,1)
            XETA=X(I,2)-X(II,2)
            YETA=Y(I,2)-Y(II,2)
            CALL PARA (XXI,YXI,XETA,YETA,LACC,CFAC,RETA,RXI,CSI,CSJ,MACC
     1,R,T,ACCL,ACCL1,NDIM,I,1,IMER,TACC,IEV)
            TEMPX=HACK2(X,T,I,1,I-1,I+1,II,2,ACCL,ACCL1,XETA,XXI)
            TEMPY=HACK2(Y,T,I,1,I-1,I+1,II,2,ACCL,ACCL1,YETA,YXI)
            R(4)=ERROR(TEMPX,X(I,1),R(4),I,1,IXER)
            R(5)=ERROR(TEMPY,Y(I,1),R(5),I,1,IYER)
            X(I,1)=TEMPX
            Y(I,1)=TEMPY
            X(II,1)=TEMPX
            Y(II,1)=TEMPY
  300       CONTINUE
          GO TO 390
C**** BOTH ON TOP
  310     DO 320 JJ=I1,I2
          I=I2-JJ+I1
            II=I4-(I-I1)
            XXI=X(I+1,JMAX)-X(I-1,JMAX)
            YXI=Y(I+1,JMAX)-Y(I-1,JMAX)
            XETA=X(II,JMXM1)-X(I,JMXM1)
            YETA=Y(II,JMXM1)-Y(I,JMXM1)
            CALL PARA (XXI,YXI,XETA,YETA,LACC,CFAC,RETA,RXI,CSI,CSJ,MACC
     1,R,T,ACCL,ACCL1,NDIM,I,JMAX,IMER,TACC,IEV)
            TEMPX=HACK3(X,T,I,JMAX,I-1,I+1,II,JMXM1,ACCL,ACCL1,XETA,XXI)
            TEMPY=HACK3(Y,T,I,JMAX,I-1,I+1,II,JMXM1,ACCL,ACCL1,YETA,YXI)
            R(4)=ERROR(TEMPX,X(I,JMAX),R(4),I,JMAX,IXER)
            R(5)=ERROR(TEMPY,Y(I,JMAX),R(5),I,JMAX,IYER)
            X(I,JMAX)=TEMPX
            Y(I,JMAX)=TEMPY
            X(II,JMAX)=TEMPX
            Y(II,JMAX)=TEMPY
  320       CONTINUE
          GO TO 390
C**** ONE ON LEFT, ONE ON RIGHT
  330     DO 340 JJ=I1,I2
          J=I2-JJ+I1
            XXI=X(2,J)-X(IMXM1,J)
            YXI=Y(2,J)-Y(IMXM1,J)
            XETA=X(1,J+1)-X(1,J-1)
            YETA=Y(1,J+1)-Y(1,J-1)
            CALL PARA (XXI,YXI,XETA,YETA,LACC,CFAC,RETA,RXI,CSI,CSJ,MACC
```

```
     1,R,T,ACCL,ACCL1,NDIM,1,J,IMER,TACC,IEV)
            TEMPX=MACK(X,T,1,J,IMXM1,2,J-1,J+1,ACCL,ACCL1,XETA,XXI)
            TEMPY=MACK(Y,T,1,J,IMXM1,2,J-1,J+1,ACCL,ACCL1,YETA,YXI)
            R(4)=ERROR(TEMPX,X(1,J),R(4),1,J,IXER)
            R(5)=ERROR(TEMPY,Y(1,J),R(5),1,J,IYER)
            X(1,J)=TEMPX
            Y(1,J)=TEMPY
            X(IMAX,J)=TEMPX
            Y(IMAX,J)=TEMPY
  340       CONTINUE
         GO TO 390
C**** BOTH ON LEFT
  350    DO 360 JJ=I1,I2
         J=I2-JJ+I1
            II=I4-(J-I1)
            XXI=X(2,J)-X(2,II)
            YXI=Y(2,J)-Y(2,II)
            XETA=X(1,J+1)-X(1,J-1)
            YETA=Y(1,J+1)-Y(1,J-1)
            CALL PARA (XXI,YXI,XETA,YETA,LACC,CFAC,RETA,RXI,CBI,CBJ,MACC
     1,R,T,ACCL,ACCL1,NDIM,1,J,IMER,TACC,IEV)
            TEMPX=MACK4(X,T,1,J,II,2,J-1,J+1,ACCL,ACCL1,XETA,XXI)
            TEMPY=MACK4(Y,T,1,J,II,2,J-1,J+1,ACCL,ACCL1,YETA,YXI)
            R(4)=ERROR(TEMPX,X(1,J),R(4),1,J,IXER)
            R(5)=ERROR(TEMPY,Y(1,J),R(5),1,J,IYER)
            X(1,J)=TEMPX
            Y(1,J)=TEMPY
            X(1,II)=TEMPX
            Y(1,II)=TEMPY
  360       CONTINUE
         GO TO 390
C**** BOTH ON RIGHT
  370    DO 380 JJ=I1,I2
         J=I2-JJ+I1
            II=I4-(J-I1)
            XXI=X(IMXM1,II)-X(IMXM1,J)
            YXI=Y(IMXM1,II)-Y(IMXM1,J)
            XETA=X(IMAX,J+1)-X(IMAX,J-1)
            YETA=Y(IMAX,J+1)-Y(IMAX,J-1)
            CALL PARA (XXI,YXI,XETA,YETA,LACC,CFAC,RETA,RXI,CBI,CBJ,MACC
     1,R,T,ACCL,ACCL1,NDIM,IMAX,J,IMER,TACC,IEV)
            TEMPX=MACK5(X,T,IMAX,J,II,IMXM1,J-1,J+1,ACCL,ACCL1,XETA,XXI)
            TEMPY=MACK5(Y,T,IMAX,J,II,IMXM1,J-1,J+1,ACCL,ACCL1,YETA,YXI)
            R(4)=ERROR(TEMPX,X(IMAX,J),R(4),IMAX,J,IXER)
            R(5)=ERROR(TEMPY,Y(IMAX,J),R(5),IMAX,J,IYER)
            X(IMAX,J)=TEMPX
            Y(IMAX,J)=TEMPY
            X(IMAX,II)=TEMPX
            Y(IMAX,II)=TEMPY
  380       CONTINUE
  390    CONTINUE
  400  CONTINUE
C
C  STORE INITIAL ITERATION ERROR NORMS
C
       R(12)=R(12)/FLOAT(NPT)
       IF (K.GT.1) GO TO 410
       R(6)=R(4)
       R(7)=R(5)
C
```

```
C   WRITE ITERATION ERROR NORMS
C
  410   IF (IWIR.GT.0) WRITE (6,590) K,R(4),R(5),R(11),R(12),LACC,LCFAC,
     1LCON
C
C   CHECK TO SEE IF ITERATION IS COMPLETE
C
C**** CHECK VARIABLE ACCELERATION PARAMETER FIELD CONVERGENCE
        LACC=R(11).LT.R(10)
        IF (LCON) LACC=.TRUE.
        IF (LACC) R(11)=0.0
C**** CHECK INTERMEDIATE FIELD CONVERGENCE BEFORE INCREASING
C**** COORDINATE ATTRACTION
        IF (LCFAC) GO TO 420
        IF (R(4).GT.EFAC.OR.R(5).GT.EFAC) GO TO 420
C**** INCREASE COORDINATE ATTRACTION
        R(6)=1.0
        LACC=.FALSE.
        IF (LCON) LACC=.TRUE.
        IF (IFAC.GT.0) CFAC=2.0*CFAC
        IF (IFAC.LT.0) CFAC=CFAC*FLOAT(IRCOUN+1)/FLOAT(IRCOUN)
        IRCOUN=IRCOUN+1
        IF (CFAC.GT.1.0) CFAC=1.0
        LCFAC=ABS(CFAC-1.0).LT.0.000001
        WRITE (6,670) CFAC
  420   CONTINUE
C**** CHECK INTERMEDIATE FIELD CONVERGENCE BEFORE MOVING
C**** OUTER BOUNDARY
      IF (INCOUN.GE.IABS(INFAC)) GO TO 430
        IF (R(4).LT.EINF.AND.R(5).LT.EINF) GO TO 490
C**** CHECK FIELD CONVERGENCE
  430   IF (R(4).LT.R(2).AND.R(5).LT.R(3)) GO TO 490
C**** PRINT VARIABLE ACCELERATION PARAMETER FIELD AT START OF ITERATION
        IF (IAIT.EQ.0) GO TO 450
        IF (K.NE.3) GO TO 450
        WRITE (6,680)
        DO 440 J=1,JMAX
          WRITE (6,690) J
          WRITE (6,700) (TACC(I,J),WACC(I,J),I=1,IMAX)
  440     CONTINUE
  450   CONTINUE
C
C   ITERATION DOES NOT CONVERGE
C
C**** WRITE PARTIALLY CONVERGED SOLUTION TO DISK
  455 WRITE (11,720) LACC,LCON,CFAC,LCFAC,IRCOUN,INCOUN,CINFAC,K,IEV,IFA
     1C,EFAC,JMXM1,IMXM1,JMXP1,CSI,CSJ,NPT,INFAC,EINF,IEND
      WRITE (11,730) (R(I),I=1,13)
      WRITE (11,730) ((X(I,J),Y(I,J),RXI(I,J),RETA(I,J),WACC(I,J),I=1,IM
     1AX),J=1,JMAX)
C**** PRINT VARIABLE ACCELERATION PARAMETER FIELD
      IF (IAIT.EQ.0) GO TO 470
      WRITE (6,680)
      DO 460 J=1,JMAX
        WRITE (6,690) J
        WRITE (6,700) (TACC(I,J),WACC(I,J),I=1,IMAX)
  460   CONTINUE
  470 CONTINUE
      IF (R(6).GT.R(4).AND.R(7).GT.R(5)) GO TO 480
      IEND=1
```

```
      RETURN
  480 IEND=2
      RETURN
C
C   ITERATION CONVERGED
C
  490 IEND=3
C**** MOVE OUTER BOUNDARY
      DINFAC=CINFAC
      IF (INCOUN.GE.IABS(INFAC)) GO TO 550
      IF (INFAC.GT.0) CINFAC=2.0*CINFAC
      IF (INFAC.LT.0) CINFAC=CINFAC*FLOAT(INCOUN+1)/FLOAT(INCOUN)
      DINFAC=CINFAC/DINFAC
      WRITE (6,710) CINFAC
      INCOUN=INCOUN+1
      DO 540 L=1,NBSEG
        I1=LB1(L)
        I2=LB2(L)
        IGOTO=LBSID(L)
        GO TO (500,520,500,520), IGOTO
  500   IF (IGOTO.EQ.1) J=1
        IF (IGOTO.EQ.3) J=JMAX
        DO 510 I=I1,I2
          X(I,J)=X(I,J)*DINFAC
  510     Y(I,J)=Y(I,J)*DINFAC
        GO TO 540
  520   IF (IGOTO.EQ.2) I=1
        IF (IGOTO.EQ.4) I=IMAX
        DO 530 J=I1,I2
          X(I,J)=X(I,J)*DINFAC
  530     Y(I,J)=Y(I,J)*DINFAC
  540   CONTINUE
      IF(LCFAC) IFAC=1
      K0=1
      IF(K.EQ.ITER) GO TO 455
      GO TO 80
C**** FINAL CONVERGENCE
  550 CONTINUE
      ITER=K
C**** PRINT VARIABLE ACCELERATION PARAMETER FIELD
      IF (IAIT.EQ.0) RETURN
      WRITE (6,680)
      DO 560 J=1,JMAX
        WRITE (6,690) J
        WRITE (6,700) (TACC(I,J),WACC(I,J),I=1,IMAX)
  560   CONTINUE
      RETURN
C
  570 FORMAT (*1*,15X,*ITERATION ERROR NORMS*//)
  580 FORMAT (I8,3E15.5,F20.5,9L10)
  590 FORMAT (2I5,2F10.0,I5)
  600 FORMAT (*0*,I5,* STEPS IN ADDITION OF INHOMOGENEOUS TERM,   INTERM
     1EDIATE CONVERGENCE FACTOR =*,F15.8)
  610 FORMAT (A6,I4,2I5,2F10.0)
  620 FORMAT (*0***** MAXIMUM NORMS OF ITERATE CHANGES*,40X,*LOGICAL CON
     1TROLS*//*0ITERATE*,9X,*X-NORM*,9X,*Y-NORM*,7X,*ACC-NORM*,7X,*AVG.A
     2CC.PARA.*,6X,*LACC*,5X,*LCFAC*,6X,*LCON*)
  630 FORMAT (*0LOGICAL CONTROLS : LACC = VARIABLE ACCELERATION*,* PARAM
     1ETER FIELD CONVERGED (IRRELEVANT IF LCON=TRUE)*//20X,*LCFAC = ADDI
     2TION OF INHOMOGENEOUS TERM COMPLETED*//20X,*LCON = UNIFORM ACCELER
```

```
     3ATION PARAMETER*//20X)
  640 FORMAT (*0COMPLEX EIGENVALUE PROCEDURE : UNDER-RELAX*)
  650 FORMAT (*0COMPLEX EIGENVALUE PROCEDURE : WEIGHTED AVERAGE*)
  660 FORMAT (*0COMPLEX EIGENVALUE PROCEDURE : OVER-RELAX*)
  670 FORMAT (*0*,F15.8,* OF INHOMOGENEOUS TERM*)
  680 FORMAT (*0ACCELERATION PARAMETERS*//)
  690 FORMAT (*0J =*,I3/)
  700 FORMAT (10(1X,*(*,I2,*)=*,F7.4,1X))
  710 FORMAT (*0*,F15.8,* OF OUTER BOUNDARY*)
  720 FORMAT(2L10,E16.8,L10,2I5,E16.8/3I5,E16.8,3I5,2E16.8/2I5,E16.8,I5)
  730 FORMAT(5E16.8)
C
      END



      SUBROUTINE PLOT(X,Y,I)
      CALL CALPLT(X,Y,I)
      RETURN
      END



      SUBROUTINE SYMBOL(X,Y,H,TEXT,ANGLE,NCHAR)
      CALL NOTATE(X,Y,H,TEXT,ANGLE,NCHAR)
      RETURN
      END
```

Sample Cases

TABLE OF CONTENTS

TABLE OF CONTENTS, CONTINUED

```
TEST CASE - BODY-FITTED COORDINATE SYSTEM
SIMPLY-CONNECTED REGION
KEY SEAT SHAFT
   21   21    0  100     0     1     1     0     0
    1    3    1    0     0     0     0     1     1
    4    0
    1    1   21   -1
    4    1   21   -1
    3   21    1   -1
    2   21    1   -1
1.8        0.0001    0.0001      0.0        0.0        0.0        0.0    0
    0    0 0.0
    0    0
8.0        0.0       10.0        0.0        0.0        0.0        0.0
 -.125      .99215
-.37461     .92718
-.62932     .77715
-.82904     .55919
-.9563      .29237
-1.
-.9563       -.29237
-.82904     -.55919
-.62932      -.77715
-.37461     -.92718
            -1.
.37461      -.92718
.62932      -.77715
.82904      -.55919
.9563       -.29237
1.
.9563       .29237
.82904       .55919
.62932      .77715
.37461       .92718
 .125       .99215
 .125       .986
 .125       .978
 .125       .97
 .125       .962
 .125       .954
 .125       .946
 .125       .936
 .125       .932
 .125       .926
 .125       .92
 .125       .914
 .125       .908
 .125       .903
 .125       .898
```

```
.125        .893
.125        .888
.125        .884
.125        .88
.125        .877
.125         .875
.12          .875
.115         .875
.11          .875
.1           .875
.09          .875
.08          .875
.06          .875
.04          .875
.02          .875
             .875
-.02         .875
-.04         .875
-.06         .875
-.08         .875
-.09         .875
-.1          .875
-.11         .875
-.115        .875
-.12         .875
-.125        .875
-.125       .877
-.125       .88
-.125       .884
-.125       .888
-.125       .893
-.125       .898
-.125       .903
-.125       .908
-.125       .914
-.125       .92
-.125       .926
-.125       .932
-.125       .936
-.125       .946
-.125       .954
-.125       .962
-.125       .97
-.125       .978
-.125       .986
ETA        0     1    0 0.0      0.0
           1  10.0     0.001
XI         0     0    0 0.0      0.0
     1     0 100.0
```

Sample Case Output: Simply-Connected Region

```
***************** INPUT ******************************

IMAX,JMAX,NBDY,ITER,IGES,IDISK,INIR,ININTL,INFIN,IGED :   21   21    0  100    0    1    1    0    0    0

IPLOT,IPLTR,NCOPY,LINWT1,LINWT2,NUMBR,NUMBR1,ISKIP1,ISKIP2 :    1    3    1    0    0    0    0    1    1

NBSEG,NRSEG :    4    0

R(1),R(2),R(3),YINFIN,AINFIN,XOINF,YOINF,NINF :
    1.80000000      .00010000      .00010000     0.00000000     0.00000000     0.00000000     0.00000000    0

IEV,IAIT,R(10) :    0    0     0.00000000

INFAC,INFACO :    0    0

SIZE,RATIO,DIST,XB1,XB2,YB1,YB2  8.00000000  0.00000000 10.00000000  0.00000000  0.00000000  0.00000000  0.00000000
*********************************************************************************************************************

SIMPLY-CONNECTED REGION

INITIAL GUESS TYPE:   0

--BODY SEGMENTS--

  L  LBSID    LB1    LB2   LBDY

  1     1      1     21     -1
  2     4      1     21     -1
  3     3     21      1     -1
  4     2     21      1     -1

--OUTER BOUNDARY--

  RADIUS =  0.00000000          INITIAL ANGLE =   0.00000000
  ORIGIN AT X = 0.00000000  , Y = 0.00000000

  NUMBER OF POINTS =    0               1 STEPS IN ATTAINMENT OF INFINITY          INITIAL STEP (LINEAR CASE) =   1

UNIFORM ACCELERATION PARAMETER USED.


                    TEST CASE - BODY-FITTED COORDINATE SYSTEM
                    SIMPLY-CONNECTED REGION

                                BODY-FITTED COORDINATE SYSTEM

                    TRANSFORMED BODY, KEY SEAT SHAFT

                    FIELD PARAMETERS, NUMBER OF XI-LINES =  21
                                      NUMBER OF ETA-LINES =   21


                    ITERATION PARAMETERS: SOR ACCELERATION PARAMETER =  1.80000
                                          MAXIMUM NUMBER OF ITERATIONS ALLOWED =  100
                                          ALLOWABLE ITERATION ERROR NORMS: X: .10000E-03
                                                                           Y: .10000E-03

                    NUMBER OF BODIES IN FIELD :    1

                    PLOT PARAMETERS:  COPIES DESIRED =   1
                                      LINEWEIGHT DESIRED =   0
                                      PLOT SIZE IN Y-DIRECTION = 8.000
                                      RATIO = 0.000

LOGICAL CONTROLS : LACC - VARIABLE ACCELERATION PARAMETER FIELD CONVERGED (IRRELEVANT IF LCON=TRUE)

                   LCFAC - ADDITION OF INHOMOGENEOUS TERM COMPLETED

                   LCON - UNIFORM ACCELERATION PARAMETER
```

EXPONENTIAL DECAY RHS
* ATTRACTION *

*** ETA EQUATION RHS ***

ATTRACTION LINES

J	AMP	DECAY
1	10.00000000	.00100000

1 STEPS IN ADDITION OF INHOMOGENEOUS TERM, INTERMEDIATE CONVERGENCE FACTOR = 100.00000000

1.00000000 OF INHOMOGENEOUS TERM

1.00000000 OF OUTER BOUNDARY

***** MAXIMUM NORMS OF ITERATE CHANGES LOGICAL CONTROLS

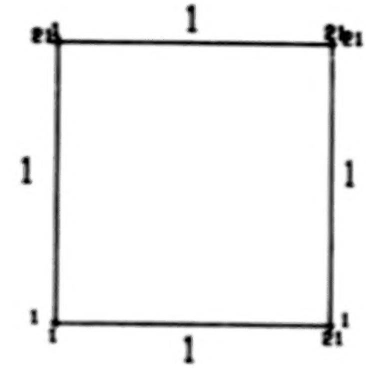
ITERATE	X-NORM	Y-NORM	ACC-NORM	AVG.ACC.PARA.	LACC	LCPAC	LCON
1	.77167E-01	.18730E+00	0.	1.00000	T	T	T
2	.73593E-01	.17935E+00	0.	1.00000	T	T	T
3	.22586E+00	.52470E+00	0.	1.80000	T	T	T
4	.19242E+00	.42021E+00	0.	1.80000	T	T	T
5	.13070E+00	.38712E+00	0.	1.80000	T	T	T
6	.10933E+00	.27774E+00	0.	1.80000	T	T	T
7	.81482E-01	.23974E+00	0.	1.80000	T	T	T
8	.75859E-01	.14885E+00	0.	1.80000	T	T	T
9	.47432E-01	.14056E+00	0.	1.80000	T	T	T
10	.47590E-01	.96423E-01	0.	1.80000	T	T	T
11	.41089E-01	.68676E-01	0.	1.80000	T	T	T
12	.29926E-01	.55841E-01	0.	1.80000	T	T	T
13	.27004E-01	.57719E-01	0.	1.80000	T	T	T
14	.17856E-01	.61240E-01	0.	1.80000	T	T	T
15	.19630E-01	.45244E-01	0.	1.80000	T	T	T
16	.19044E-01	.29517E-01	0.	1.80000	T	T	T
17	.17046E-01	.32743E-01	0.	1.80000	T	T	T
18	.14024E-01	.40620E-01	0.	1.80000	T	T	T
19	.89439E-02	.33133E-01	0.	1.80000	T	T	T
20	.99048E-02	.26755E-01	0.	1.80000	T	T	T
21	.10402E-01	.22490E-01	0.	1.80000	T	T	T
22	.11177E-01	.20057E-01	0.	1.80000	T	T	T
23	.90140E-02	.16369E-01	0.	1.80000	T	T	T
24	.71066E-02	.12976E-01	0.	1.80000	T	T	T
25	.56684E-02	.10755E-01	0.	1.80000	T	T	T
26	.41851E-02	.80856E-02	0.	1.80000	T	T	T
27	.33512E-02	.64339E-02	0.	1.80000	T	T	T
28	.26513E-02	.44527E-02	0.	1.80000	T		T
29	.21974E-02	.32657E-02	0.	1.80000	T	T	T
30	.19004E-02	.25563E-02	0.	1.80000	T	T	T
31	.14068E-02	.17945E-02	0.	1.80000	T	T	T
32	.10214E-02	.10352E-02	0.	1.80000	T	T	T
33	.76642E-03	.69375E-03	0.	1.80000	T	T	T
34	.62992E-03	.83619E-03	0.	1.80000	T	T	T
35	.49023E-03	.53767E-03	0.	1.80000	T	T	T
36	.47594E-03	.42585E-03	0.	1.80000	T	T	T
37	.28581E-03	.36137E-03	0.	1.80000	T	T	T
38	.36817E-03	.39204E-03	0.	1.80000	T	T	T
39	.32856E-03	.30726E-03	0.	1.80000	T	T	T
40	.27069E-03	.31306E-03	0.	1.80000	T	T	T
41	.18414E-03	.26687E-03	0.	1.80000	T	T	T
42	.14048E-03	.28925E-03	0.	1.80000	T	T	T
43	.98688E-04	.19891E-03	0.	1.80000	T	T	T
44	.11059E-03	.14287E-03	0.	1.80000	T	T	T
45	.13414E-03	.14653E-03	0.	1.80000	T	T	T
46	.55074E-04	.78366E-04	0.	1.80000	T	T	T

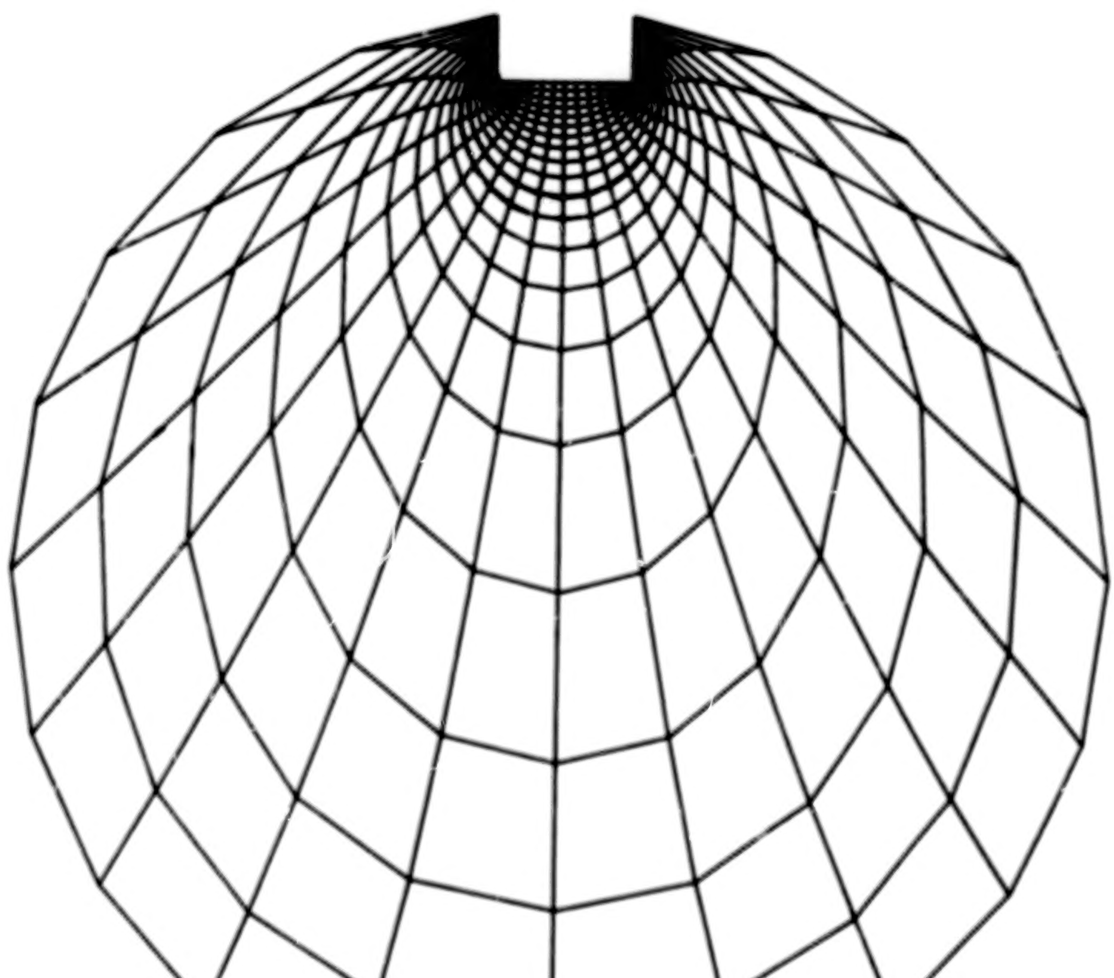
```
TEST CASE - BODY-FITTED COORDINATE SYSTEM
SIMPLY-CONNECTED REGION

                              FINAL VALUES

ITERATION CONVERGED.

INITIAL ITERATION ERROR NORMS:  X: .77167E-01  Y: .18730E+00 AT ITERATE # 1
FINAL ITERATION ERROR NORMS:    X: .55074E-04  Y: .78360E-04 AT ITERATE # 46
LOCATION OF MAXIMUM ITERATION ERROR: X: I=    5,J=    4
                                     Y: I=   17 J=    5

*********************************        X-ARRAY        ******************************************************
 J=   1
-.12500E+00-.37461E+00-.62932E+00-.82904E+00-.95630E+00-.10000E+01-.95630E+00-.82904E+00-.62932E+00-.37461E+00
0.          .37461E+00 .62932E+00 .82904E+00 .95630E+00 .10000E+01 .95630E+00 .82904E+00 .62932E+00 .37461E+00
 .12500E+00
 J=   2
-.12500E+00-.28715E+00-.46551E+00-.63666E+00-.76855E+00-.83593E+00-.82204E+00-.72549E+00-.55131E+00-.31209E+00
-.41931E-05 .31288E+00 .55130E+00 .72547E+00 .82201E+00 .83590E+00 .76852E+00 .63661E+00 .46551E+00 .28714E+00
 .12500E+00
 J=   3
-.12500E+00-.24193E+00-.36117E+00-.48457E+00-.59338E+00-.66367E+00-.67305E+00-.60747E+00-.46592E+00-.25066E+00
-.11970E-04 .25058E+00 .46569E+00 .60742E+00 .67297E+00 .66356E+00 .59331E+00 .48451E+00 .36116E+00 .24191E+00
 .12500E+00
 J=   4
-.12500E+00-.21621E+00-.29906E+00-.38120E+00-.45569E+00-.50838E+00-.52150E+00-.47076E+00-.37191E+00-.20572E+00
-.16426E-04 .20569E+00 .37186E+00 .47067E+00 .52130E+00 .50825E+00 .45562E+00 .38110E+00 .29905E+00 .21610E+00
 .12500E+00
 J=   5
-.12500E+00-.19937E+00-.26073E+00-.31592E+00-.36207E+00-.39170E+00-.39463E+00-.35911E+00-.27016E+00-.15328E+00
-.60858E-05 .15325E+00 .27012E+00 .35903E+00 .39433E+00 .39162E+00 .36201E+00 .31592E+00 .26071E+00 .19933E+00
 .12500E+00
 J=   6
-.12500E+00-.18707E+00-.23457E+00-.27328E+00-.30155E+00-.31472E+00-.30671E+00-.27155E+00-.20574E+00-.11164E+00
-.17932E-05 .11163E+00 .20571E+00 .27151E+00 .30668E+00 .31471E+00 .30154E+00 .27326E+00 .23456E+00 .18703E+00
 .12500E+00
 J=   7
-.12500E+00-.17743E+00-.21505E+00-.24290E+00-.26022E+00-.26391E+00-.25019E+00-.21590E+00-.16005E+00-.85540E-01
 .82325E-05 .85544E-01 .16004E+00 .21589E+00 .25018E+00 .26390E+00 .26022E+00 .23288E+00 .21502E+00 .17741E+00
 .12500E+00
 J=   8
-.12500E+00-.16955E+00-.19959E+00-.21968E+00-.22972E+00-.22775E+00-.21137E+00-.17895E+00-.13050E+00-.69051E-01
 .20248E-05 .69065E-01 .13058E+00 .17896E+00 .21137E+00 .22775E+00 .22972E+00 .21967E+00 .19956E+00 .16953E+00
 .12500E+00
 J=   9
-.12500E+00-.16300E+00-.18689E+00-.20110E+00-.20598E+00-.20034E+00-.18276E+00-.15243E+00-.10992E+00-.57603E-01
 .89007E-05 .57697E-01 .10993E+00 .15243E+00 .18275E+00 .20034E+00 .20597E+00 .20109E+00 .18687E+00 .16290E+00
 .12500E+00
 J=  10
-.12500E+00-.15736E+00-.17616E+00-.18576E+00-.18680E+00-.17870E+00-.16063E+00-.13233E+00-.94525E-01-.49301E-01
 .86594E-05 .49314E-01 .94635E-01 .13233E+00 .16065E+00 .17871E+00 .18682E+00 .18574E+00 .17619E+00 .15732E+00
 .12500E+00
 J=  11
-.12500E+00-.15239E+00-.16691E+00-.17279E+00-.17092E+00-.16111E+00-.14298E+00-.11653E+00-.82575E-01-.42852E-01
 .47050E-05 .42867E-01 .82563E-01 .11656E+00 .14300E+00 .16114E+00 .17094E+00 .17279E+00 .16691E+00 .15236E+00
 .12500E+00
 J=  12
-.12500E+00-.14795E+00-.15886E+00-.16164E+00-.15754E+00-.14656E+00-.12860E+00-.10386E+00-.73069E-01-.37790E-01
 .70007E-05 .37757E-01 .73082E-01 .10386E+00 .12863E+00 .14659E+00 .15757E+00 .16167E+00 .15882E+00 .14790E+00
 .12500E+00
```

```
J= 13
-.12500E+00-.14394E+00-.15160E+00-.15193E+00-.14611E+00-.13436E+00-.11673E+00-.83493E-01-.65391E-01-.33697E-01
-.23812E-05 .33667E-01 .65394E-01 .93512E-01 .11676E+00 .13440E+00 .14616E+00 .15198E+00 .15162E+00 .14394E+00
 .12500E+00
J= 14
-.12500E+00-.14029E+00-.14514E+00-.14341E+00-.13629E+00-.12408E+00-.10660E+00-.85019E-01-.59142E-01-.30353E-01
 .46914E-05 .30349E-01 .59157E-01 .85039E-01 .10692E+00 .12412E+00 .13634E+00 .14347E+00 .14518E+00 .14030E+00
 .12500E+00
J= 15
-.12500E+00-.13642E+00-.13931E+00-.13591E+00-.12763E+00-.11539E+00-.98727E-01-.78071E-01-.54078E-01-.27662E-01
 .65146E-05 .27674E-01 .54043E-01 .78100E-01 .98763E-01 .11544E+00 .12769E+00 .13596E+00 .13935E+00 .13694E+00
 .12500E+00
J= 16
-.12500E+00-.13376E+00-.13402E+00-.12930E+00-.12058E+00-.10812E+00-.92024E-01-.72453E-01-.49982E-01-.25507E-01
 .71054E-05 .25516E-01 .50005E-01 .72473E-01 .92064E-01 .10817E+00 .12063E+00 .12936E+00 .13406E+00 .13380E+00
 .12500E+00
J= 17
-.12500E+00-.13062E+00-.12923E+00-.12353E+00-.11442E+00-.10213E+00-.86666E-01-.67985E-01-.46745E-01-.23795E-01
 .62447E-05 .23966E-01 .46763E-01 .68010E-01 .86697E-01 .10217E+00 .11447E+00 .12358E+00 .12926E+00 .13064E+00
 .12500E+00
J= 18
-.12500E+00-.12804E+00-.12490E+00-.11859E+00-.10932E+00-.97349E-01-.82580E-01-.64597E-01-.44246E-01-.22461E-01
 .47674E-05 .22465E-01 .44259E-01 .64614E-01 .82606E-01 .97379E-01 .10936E+00 .11862E+00 .12492E+00 .12805E+00
 .12500E+00
J= 19
-.12500E+00-.12537E+00-.12103E+00-.11452E+00-.10526E+00-.93749E-01-.79852E-01-.62209E-01-.42306E-01-.21435E-01
 .11030E-06 .21436E-01 .42393E-01 .62219E-01 .79864E-01 .93768E-01 .10528E+00 .11454E+00 .12104E+00 .12538E+00
 .12500E+00
J= 20
-.12500E+00-.12276E+00-.11768E+00-.11153E+00-.10221E+00-.91330E-01-.76733E-01-.60755E-01-.41036E-01-.20651E-01
-.11656E-06 .20646E-01 .41041E-01 .60760E-01 .76734E-01 .91336E-01 .10222E+00 .11154E+00 .11768E+00 .12276E+00
 .12500E+00
J= 21
-.12500E+00-.12000E+00-.11500E+00-.11000E+00-.10000E+00-.90000E-01-.80000E-01-.60000E-01-.40000E-01-.20000E-01
0.          .20000E-01 .40000E-01 .60000E-01 .80000E-01 .90000E-01 .10000E+00 .11000E+00 .11500E+00 .12000E+00
 .12500E+00

 ********************************        Y=ARRAY       *******************,****************************
J= 1
 .99215E+00 .92718E+00 .77715E+00 .55919E+00 .29237E+000.          -.29237E+00-.55919E+00-.77715E+00-.92718E+00
-.10000E+01-.92718E+00-.77715E+00-.55919E+00-.29237E+000.           .29237E+00 .55919E+00 .77715E+00 .92718E+00
 .99215E+00
J= 2
 .98600E+00 .93706E+00 .82634E+00 .65070E+00 .41056E+00 .14921E+00-.13025E+00-.39042E+00-.60492E+00-.75304E+00
-.81400E+00-.75304E+00-.60490E+00-.39039E+00-.13022E+00 .14925E+00 .41061E+00 .65074E+00 .82642E+00 .93707E+00
 .98600E+00
J= 3
 .97600E+00 .93097E+00 .85767E+00 .72515E+00 .53959E+00 .30974E+00 .56384E-01-.19057E+00-.39940E+00-.54295E+00
-.59675E+00-.54294E+00-.39934E+00-.19046E+00 .56511E-01 .30989E+00 .53970E+00 .72524E+00 .85770E+00 .93098E+00
 .97600E+00
J= 4
 .97000E+00 .93636E+00 .87367E+00 .77539E+00 .63660E+00 .45885E+00 .25201E+00 .38149E-01-.15204E+00-.28570E+00
-.33452E+00-.28565E+00-.15193E+00 .38334E-01 .25226E+00 .45907E+00 .63696E+00 .77567E+00 .87370E+00 .93640E+00
 .97000E+00
J= 5
 .96200E+00 .93188E+00 .88035E+00 .80469E+00 .70215E+00 .57260E+00 .42108E+00 .26060E+00 .11327E+00 .71993E-02
-.31776E-01 .72859E-02 .11344E+00 .26084E+00 .42134E+00 .57283E+00 .70231E+00 .80473E+00 .88039E+00 .93190E+00
 .96200E+00
J= 6
 .95400E+00 .92662E+00 .88244E+00 .82113E+00 .74206E+00 .64662E+00 .53090E+00 .42703E+00 .32734E+00 .25636E+00
 .23049E+00 .25641E+00 .32750E+00 .42764E+00 .53074E+00 .64695E+00 .74214E+00 .82126E+00 .88246E+00 .92608E+00
 .95400E+00
```

```
J=  7
.94600E+00 .92106E+00 .88233E+00 .83093E+00 .76726E+00 .69319E+00 .61287E+00 .53329E+00 .46605E+00 .41630E+00
.39922E+00 .41636E+00 .46617E+00 .53336E+00 .61297E+00 .69326E+00 .76730E+00 .83097E+00 .88230E+00 .92118E+00
.94600E+00
J=  8
.93600E+00 .91544E+00 .88109E+00 .83717E+00 .78453E+00 .72515E+00 .66265E+00 .60254E+00 .55171E+00 .51742E+00
.50524E+00 .51747E+00 .55178E+00 .60259E+00 .66271E+00 .72517E+00 .78450E+00 .83721E+00 .88116E+00 .91545E+00
.93600E+00
J=  9
.93200E+00 .91002E+00 .87929E+00 .84136E+00 .79720E+00 .74861E+00 .69868E+00 .65169E+00 .61275E+00 .58605E+00
.57777E+00 .58687E+00 .61278E+00 .65173E+00 .69870E+00 .74866E+00 .79724E+00 .84143E+00 .87930E+00 .91005E+00
.93200E+00
J= 10
.92600E+00 .90485E+00 .87723E+00 .84432E+00 .80700E+00 .76683E+00 .72633E+00 .68885E+00 .65824E+00 .63818E+00
.63107E+00 .63811E+00 .65826E+00 .68887E+00 .72635E+00 .76686E+00 .80703E+00 .84435E+00 .87725E+00 .90490E+00
.92600E+00
J= 11
.92000E+00 .89992E+00 .87512E+00 .84656E+00 .81497E+00 .78162E+00 .74853E+00 .71832E+00 .69391E+00 .67799E+00
.67245E+00 .67799E+00 .69341E+00 .71832E+00 .74852E+00 .78163E+00 .81499E+00 .84658E+00 .87517E+00 .89997E+00
.92000E+00
J= 12
.91400E+00 .89525E+00 .87310E+00 .84842E+00 .82175E+00 .79410E+00 .76703E+00 .74259E+00 .72301E+00 .71031E+00
.70591E+00 .71030E+00 .72300E+00 .74256E+00 .76699E+00 .79408E+00 .82175E+00 .84844E+00 .87316E+00 .89530E+00
.91400E+00
J= 13
.90800E+00 .89069E+00 .87126E+00 .85011E+00 .82778E+00 .80499E+00 .78294E+00 .76320E+00 .74748E+00 .73734E+00
.73384E+00 .73733E+00 .74746E+00 .76316E+00 .78290E+00 .80497E+00 .82779E+00 .85015E+00 .87131E+00 .89094E+00
.90800E+00
J= 14
.90300E+00 .88697E+00 .86969E+00 .85183E+00 .83338E+00 .81481E+00 .79701E+00 .78117E+00 .76863E+00 .76056E+00
.75777E+00 .76054E+00 .76859E+00 .78113E+00 .79698E+00 .81481E+00 .83339E+00 .85187E+00 .86974E+00 .88693E+00
.90300E+00
J= 15
.89800E+00 .88320E+00 .86846E+00 .85371E+00 .83877E+00 .82392E+00 .80978E+00 .79724E+00 .78732E+00 .78096E+00
.77876E+00 .78094E+00 .78728E+00 .79719E+00 .80975E+00 .82391E+00 .83878E+00 .85375E+00 .86853E+00 .88325E+00
.89800E+00
J= 16
.89300E+00 .87995E+00 .86772E+00 .85590E+00 .84415E+00 .83259E+00 .82163E+00 .81189E+00 .80420E+00 .79927E+00
.79757E+00 .79925E+00 .80416E+00 .81185E+00 .82160E+00 .83258E+00 .84415E+00 .85592E+00 .86775E+00 .87990E+00
.89300E+00
J= 17
.88800E+00 .87721E+00 .86751E+00 .85849E+00 .84966E+00 .84104E+00 .83286E+00 .82553E+00 .81973E+00 .81602E+00
.81475E+00 .81601E+00 .81970E+00 .82549E+00 .83282E+00 .84101E+00 .84965E+00 .85050E+00 .86752E+00 .87722E+00
.88800E+00
J= 18
.88400E+00 .87511E+00 .86796E+00 .86162E+00 .85542E+00 .84941E+00 .84368E+00 .83846E+00 .83429E+00 .83165E+00
.83075E+00 .83165E+00 .83427E+00 .83842E+00 .84365E+00 .84939E+00 .85541E+00 .86162E+00 .86796E+00 .87512E+00
.88400E+00
J= 19
.88000E+00 .87379E+00 .86924E+00 .86539E+00 .86153E+00 .85781E+00 .85427E+00 .85087E+00 .84818E+00 .84649E+00
.84592E+00 .84649E+00 .84817E+00 .85085E+00 .85426E+00 .85780E+00 .86151E+00 .86538E+00 .86923E+00 .87379E+00
.88000E+00
J= 20
.87700E+00 .87359E+00 .87153E+00 .86987E+00 .86803E+00 .86633E+00 .86474E+00 .86300E+00 .86167E+00 .86085E+00
.86056E+00 .86085E+00 .86166E+00 .86294E+00 .86472E+00 .86633E+00 .86802E+00 .86986E+00 .87152E+00 .87350E+00
.87700E+00
J= 21
.87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00
.87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00 .87500E+00
.87500E+00
```

Sample Case Plot: Simply Connected Region





Sample Case Input: Single-Body Field

```
TEST CASE - BODY-FITTED COORDINATE SYSTEM
SINGLE BODY : KARMAN-TREFFTZ AIRFOIL, 26 POINTS
K-T AIRFOIL #1
   27   20     1  200     0     1     1     0     0
    1    3     1    0     0     0     0     1     1
    2    1
    1    1    27   -1
    3    1    27    0
    2    1    20    4     1    20
1.8          0.0001     0.0001     5.0        0.0        0.0        0.0        26
    0    0 0.0
    0    0
8.0          0.0        10.0       0.0        0.0        0.0        0.0
   .499528  -.000031
   .498623  -.001402
   .496086  -.002760
   .486293  -.005552
   .470464  -.008509
   .390148  -.018591
   .269552  -.029966
   .122707  -.040790
  -.034802  -.048453
  -.187472  -.050524
  -.321146  -.045592
  -.423886  -.033739
  -.486506  -.016526
  -.502726   .003820
  -.471102   .026435
  -.396903   .048642
  -.287913   .066101
  -.153756   .075299
  -.005644   .074531
   .144046   .064352
   .282251   .047498
   .345776   .028262
   .471990   .011299
   .487103   .006809
   .496419   .003040
   .498779   .001426
ETA        0      4     2 0.0        0.0
           1 100.0     1.0
           2 100.0     1.0
           3 100.0     1.0
           4 100.0     1.0
    1      1 100.0     1.0
   27      1 100.0     1.0
XI         0      0     0 0.0        0.0
    1      0 100.0
```

Sample Case Output: Single-Body Field

[illegible]

```
TEST CASE = MODIFIED COORDINATE SYSTEM
SINGLE BODY : KARMAN-TREFFTZ AIRFOIL, 26 POINTS

                 MODIFIED COORDINATE SYSTEM

TRANSFORMED BODY, K-T AIRFOIL #1

FIELD PARAMETERS, NUMBER OF XI-LINES =  27
                  NUMBER OF ETA-LINES =   26


ITERATION PARAMETERS: SOR ACCELERATION PARAMETER =  1.00000
                      MAXIMUM NUMBER OF ITERATIONS ALLOWED =  200
                      ALLOWABLE ITERATION ERROR NORMS: X: .10000E-03
                                                      Y: .10000E-03

NUMBER OF BODIES IN FIELD :    1

PLOT PARAMETERS:  COPIES DESIRED =   1
                  LINEWEIGHT DESIRED =    0
                  PLOT SIZE IN Y-DIRECTION = 0.000
                  RATIO = 0.000

LOGICAL CONTROLS : LACC = VARIABLE ACCELERATION PARAMETER FIELD CONVERGED (IRRELEVANT IF LCON=TRUE)

                   LCFAC = ADDITION OF INHOMOGENEOUS TERM COMPLETED

                   LCON = UNIFORM ACCELERATION PARAMETER


EXPONENTIAL DECAY RHS
- ATTRACTION -

*** ETA EQUATION RHS ***

ATTRACTION LINES

   J            AMP                 DECAY
   1       100.00000000         1.00000000
   2       100.00000000         1.00000000
   3       100.00000000         1.00000000
   4       100.00000000         1.00000000

ATTRACTION POINTS

   I    J               AMP                 DECAY   LPT
   1    1         100.00000000         1.00000000    0
  27    1         100.00000000         1.00000000    0

   1 STEPS IN ADDITION OF INHOMOGENEOUS TERM,   INTERMEDIATE CONVERGENCE FACTOR =   100.00000000

    1.00000000 OF INHOMOGENEOUS TERM

    1.00000000 OF OUTER BOUNDARY

***** MAXIMUM NORMS OF ITERATE CHANGES                                    LOGICAL CONTROLS


ITERATE        XNORM          YNORM        ACCNORM        AVG.ACC.PARA.     LACC      LCFAC      LCON
     1      .14090E+01     .36062E+00    0.                    1.00000        T         T          T
     2      .72723E+00     .34206E+00    0.                    1.00000        T         T          T
     3      .11796E+01     .10532E+01    0.                    1.00000        T         T          T
     4      .94555E+00     .33196E+00    0.                    1.00000        T         T          T
     5      .27625E+00     .17067E+00    0.                    1.00000        T         T          T
```

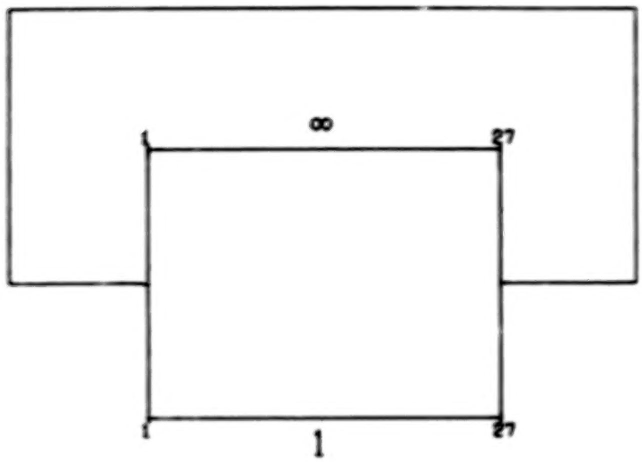
6	.27424E+00	.15720E+00	0.	1.00000	T	T	T
7	.25074E+00	.16559E+00	0.	1.00000	T	T	T
8	.23441E+00	.16888E+00	0.	1.00000	T	T	T
9	.23336E+00	.16002E+00	0.	1.00000	T	T	T
10	.21302E+00	.16367E+00	0.	1.00000	T	T	T
11	.22400E+00	.17011E+00	0.	1.00000	T	T	T
12	.21953E+00	.16540E+00	0.	1.00000	T	T	T
13	.20770E+00	.15442E+00	0.	1.00000	T	T	T
14	.19429E+00	.16858E+00	0.	1.00000	T	T	T
15	.18325E+00	.15459E+00	0.	1.00000	T	T	T
16	.15600E+00	.12500E+00	0.	1.00000	T	T	T
17	.11877E+00	.83332E-01	0.	1.00000	T	T	T
18	.82235E-01	.81247E-01	0.	1.00000	T	T	T
19	.64074E-01	.78787E-01	0.	1.00000	T	T	T
20	.10143E+00	.92858E-01	0.	1.00000	T	T	T
21	.94124E-01	.76010E-01	0.	1.00000	T	T	T
22	.94005E-01	.65372E-01	0.	1.00000	T	T	T
23	.84595E-01	.79466E-01	0.	1.00000	T	T	T
24	.82749E-01	.80437E-01	0.	1.00000	T	T	T
25	.49460E-01	.75932E-01	0.	1.00000	T	T	T
26	.57533E-01	.67772E-01	0.	1.00000	T	T	T
27	.58655E-01	.72037E-01	0.	1.00000	T	T	T
28	.51078E-01	.65519E-01	0.	1.00000	T	T	T
29	.44573E-01	.52442E-01	0.	1.00000	T	T	T
30	.57314E-01	.44049E-01	0.	1.00000	T	T	T
31	.54161E-01	.50925E-01	0.	1.00000	T	T	T
32	.38008E-01	.52878E-01	0.	1.00000	T	T	T
33	.34888E-01	.51284E-01	0.	1.00000	T	T	T
34	.39245E-01	.42320E-01	0.	1.00000	T	T	T
35	.30429E-01	.38540E-01	0.	1.00000	T	T	T
36	.44415E-01	.34704E-01	0.	1.00000	T	T	T
37	.42778E-01	.35775E-01	0.	1.00000	T	T	T
38	.40438E-01	.23407E-01	0.	1.00000	T	T	T
39	.37437E-01	.13114E-01	0.	1.00000	T	T	T
40	.26262E-01	.68941E-02	0.	1.00000	T	T	T
41	.10522E-01	.44241E-02	0.	1.00000	T	T	T
42	.11689E-01	.51653E-02	0.	1.00000	T	T	T
43	.53887E-02	.49058E-02	0.	1.00000	T	T	T
44	.55185E-02	.37144E-02	0.	1.00000	T	T	T
45	.35818E-02	.39505E-02	0.	1.00000	T	T	T
46	.25905E-02	.36130E-02	0.	1.00000	T	T	T
47	.15855E-02	.25742E-02	0.	1.00000	T	T	T
48	.10572E-02	.16755E-02	0.	1.00000	T	T	T
49	.94412E-03	.10031E-02	0.	1.00000	T	T	T
50	.75459E-03	.77094E-03	0.	1.00000	T	T	T
51	.64234E-03	.60310E-03	0.	1.00000	T	T	T
52	.52780E-03	.53440E-03	0.	1.00000	T	T	T
53	.43784E-03	.44348E-03	0.	1.00000	T	T	T
54	.39497E-03	.35394E-03	0.	1.00000	T	T	T
55	.30279E-03	.31222E-03	0.	1.00000	T	T	T
56	.23031E-03	.28355E-03	0.	1.00000	T	T	T
57	.17911E-03	.23353E-03	0.	1.00000	T	T	T
58	.16377E-03	.20308E-03	0.	1.00000	T	T	T
59	.17508E-03	.15065E-03	0.	1.00000	T	T	T
60	.12818E-03	.15035E-03	0.	1.00000	T	T	T
61	.13322E-03	.13840E-03	0.	1.00000	T	T	T
62	.94475E-04	.12497E-03	0.	1.00000	T	T	T
63	.81150E-04	.10813E-03	0.	1.00000	T	T	T
64	.10981E-03	.93708E-04	0.	1.00000	T	T	T
65	.87793E-04	.93496E-04	0.	1.00000	T	T	T

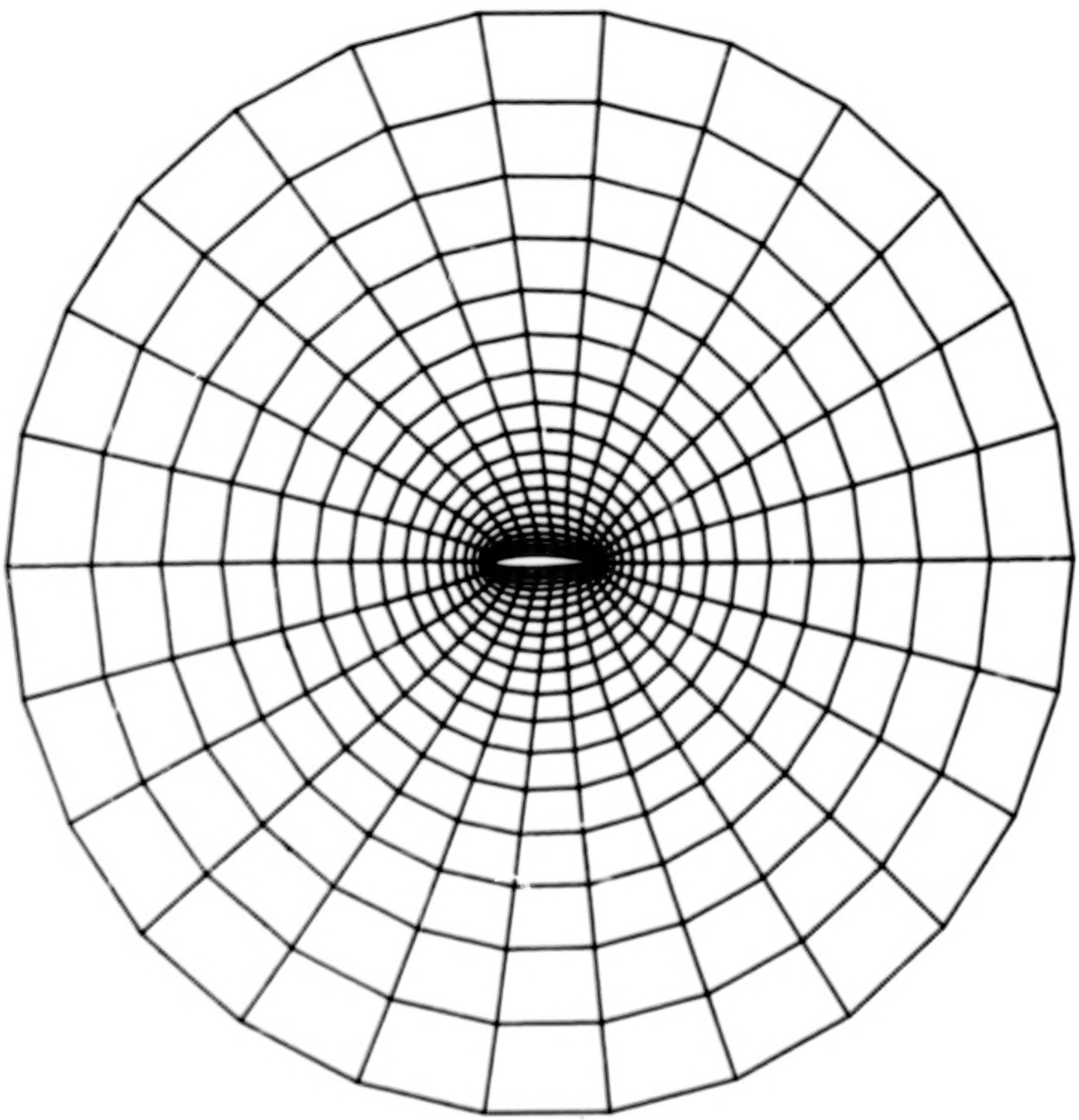
```
TEST CASE - BODY-FITTED COORDINATE SYSTEM
SINGLE BODY : KARMAN-TREFFTZ AIRFOIL, 26 POINTS

                              FINAL VALUES

ITERATION CONVERGES,

INITIAL ITERATION ERROR NORMS:  X: ,14090E+01  Y: ,36662E+00 AT ITERATE # 1
FINAL ITERATION ERROR NORMS:    X: ,67793E-04  Y: ,93298E-04 AT ITERATE #  65
LOCATION OF MAXIMUM ITERATION ERROR: X: I=   11,J=   19
                                     Y: I=   25 J=   17

*********************************          X-ARRAY        *********************,******************************
 J=   1
 ,49953E+00 ,49662E+00 ,49609E+00 ,48629E+00 ,47046E+00 ,39015E+00 ,26955E+00 ,12271E+00-,34802E-01-,18747E+00
-,32115E+00-,42369E+00-,48651E+00-,50273E+00-,47110E+00-,39690E+00-,26791E+00-,15576E+00-,56460E-02 ,14405E+00
 ,28225E+00 ,39578E+00 ,47199E+00 ,48710E+00 ,49642E+00 ,49878E+00 ,49953E+00
 J=   2
 ,50647E+00 ,50430E+00 ,49742E+00 ,48143E+00 ,46602E+00 ,36743E+00 ,25068E+00 ,10776E+00-,46750E-01-,19756E+00
-,33036E+00-,43295E+00-,49585E+00-,51278E+00-,48207E+00-,40643E+00-,29959E+00-,16559E+00-,18128E-01 ,13602E+00
 ,26564E+00 ,37611E+00 ,45134E+00 ,48415E+00 ,49894E+00 ,50502E+00 ,50647E+00
 J=   3
 ,52006E+00 ,51622E+00 ,50465E+00 ,48071E+00 ,43495E+00 ,35261E+00 ,23650E+00 ,94971E-01-,58467E-01-,20905E+00
-,34255E+00-,44647E+00-,51076E+00-,52662E+00-,49814E+00-,42376E+00-,31366E+00-,17647E+00-,30288E-01 ,11797E+00
 ,25534E+00 ,36413E+00 ,44222E+00 ,48552E+00 ,50762E+00 ,51764E+00 ,52006E+00
 J=   4
 ,54069E+00 ,53481E+00 ,51797E+00 ,48586E+00 ,43069E+00 ,34368E+00 ,22537E+00 ,83192E-01-,70847E-01-,22272E+00
-,35843E+00-,46513E+00-,53187E+00-,55095E+00-,52005E+00-,44374E+00-,33101E+00-,19327E+00-,43067E-01 ,10689E+00
 ,24393E+00 ,35740E+00 ,44052E+00 ,49273E+00 ,52243E+00 ,53700E+00 ,54069E+00
 J=   5
 ,57017E+00 ,56198E+00 ,53919E+00 ,49816E+00 ,43322E+00 ,33098E+00 ,21067E+00 ,70816E-01-,85095E-01-,24095E+00
-,38060E+00-,49141E+00-,56137E+00-,58185E+00-,55019E+00-,47109E+00-,35039E+00-,21244E+00-,58255E-01 ,95331E-01
 ,23660E+00 ,35514E+00 ,44562E+00 ,50725E+00 ,54520E+00 ,56490E+00 ,57017E+00
 J=   6
 ,61255E+00 ,60176E+00 ,57225E+00 ,52121E+00 ,44478E+00 ,34008E+00 ,20853E+00 ,56200E-01-,10670E+00-,26795E+00
-,41369E+00-,52983E+00-,60369E+00-,62540E+00-,59297E+00-,51059E+00-,38062E+00-,24018E+00-,79002E-01 ,81912E-01
 ,23096E+00 ,35888E+00 ,45967E+00 ,53262E+00 ,57992E+00 ,60562E+00 ,61255E+00
 J=   7
 ,67078E+00 ,65709E+00 ,62005E+00 ,55766E+00 ,46709E+00 ,34916E+00 ,20458E+00 ,40394E-01-,13357E+00-,30493E+00
-,45958E+00-,58274E+00-,66093E+00-,68468E+00-,65101E+00-,56443E+00-,43538E+00-,27759E+00-,10549E+00 ,67362E-01
 ,22918E+00 ,37059E+00 ,48544E+00 ,57137E+00 ,62936E+00 ,66181E+00 ,67078E+00
 J=   8
 ,74770E+00 ,73076E+00 ,68512E+00 ,60967E+00 ,50352E+00 ,36736E+00 ,20491E+00 ,23378E-01-,16688E+00-,35290E+00
-,51987E+00-,65234E+00-,73636E+00-,76224E+00-,72690E+00-,63472E+00-,49623E+00-,32561E+00-,13810E+00 ,51852E-01
 ,23168E+00 ,39132E+00 ,52380E+00 ,62555E+00 ,69604E+00 ,73632E+00 ,74770E+00
 J=   9
 ,84620E+00 ,82557E+00 ,76994E+00 ,67909E+00 ,55343E+00 ,39522E+00 ,20957E+00 ,68021E-02-,20736E+00-,41308E+00
-,59638E+00-,74101E+00-,83255E+00-,86107E+00-,82341E+00-,72382E+00-,57295E+00-,38550E+00-,17764E+00 ,34969E-01
 ,23648E+00 ,42161E+00 ,57612E+00 ,69705E+00 ,78239E+00 ,83193E+00 ,84620E+00
 J=  10
 ,96956E+00 ,94462E+00 ,87721E+00 ,76798E+00 ,61872E+00 ,43327E+00 ,21844E+00-,15735E-01-,25615E+00-,48712E+00
-,69134E+00-,85152E+00-,95263E+00-,98446E+00-,94309E+00-,83453E+00-,66788E+00-,45902E+00-,22534E+00 ,16002E-01
 ,24944E+00 ,46196E+00 ,64365E+00 ,78791E+00 ,89110E+00 ,95175E+00 ,96956E+00
 J=  11
 ,11216E+01 ,10916E+01 ,10102E+01 ,87890E+00 ,70105E+00 ,48233E+00 ,23151E+00-,39268E-01-,31476E+00-,57720E+00
-,80756E+00-,98722E+00-,11003E+01-,11363E+01-,10915E+01-,97030E+00-,78309E+00-,54848E+00-,28280E+00-,59209E-02
 ,26446E+00 ,51314E+00 ,72807E+00 ,90062E+00 ,10254E+01 ,10994E+01 ,11216E+01
```

```
J=  12
 .15064E+01 .12710E+01 .11728E+01 .10151E+01 .80271E+00 .54357E+00 .24891E+00-.60010E-01-.38501E+00-.60607E+00
-.94055E+00-.11523E+01-.12802E+01-.13212E+01-.12714E+01-.11353E+01-.92407E+00-.65075E+00-.35203E+00-.31037E-01
 .28355E+00 .57624E+00 .83152E+00 .10364E+01 .11692E+01 .12794E+01 .15064E+01
J=  13
 .15512E+01 .14443E+01 .13701E+01 .11605E+01 .92658E+00 .61607E+00 .27091E+00-.08754E-01-.40917E+00-.61707E+00
-.11187E+01-.14517E+01-.14470E+01-.15450E+01-.14891E+01-.13348E+01-.10947E+01-.78737E+00-.43546E+00-.62917E-01
 .30647E+00 .65275E+00 .95681E+00 .12051E+01 .13873E+01 .14972E+01 .15512E+01
J=  14
 .18013E+01 .17502E+01 .16606E+01 .13803E+01 .10764E+01 .70968E+00 .29801E+00-.13687E+00-.56994E+00-.97433E+00
-.13252E+01-.15917E+01-.17498E+01-.18144E+01-.17512E+01-.15750E+01-.12994E+01-.94470E+00-.53605E+00-.10659E+00
 .33464E+00 .74454E+00 .11074E+01 .14055E+01 .16259E+01 .17594E+01 .18013E+01
J=  15
 .21259E+01 .20644E+01 .18946E+01 .16205E+01 .12565E+01 .81921E+00 .33078E+00-.18236E+00-.69063E+00-.11624E+01
-.15665E+01-.18797E+01-.20743E+01-.21380E+01-.20660E+01-.18630E+01-.15454E+01-.11340E+01-.65744E+00-.14659E+00
 .38730E+00 .85345E+00 .12874E+01 .16450E+01 .19114E+01 .20742E+01 .21259E+01
J=  16
 .25149E+01 .24423E+01 .22370E+01 .19064E+01 .14729E+01 .95636E+00 .38993E+00-.23099E+00-.83536E+00-.13889E+01
-.18623E+01-.22247E+01-.24511E+01-.25250E+01-.24434E+01-.22097E+01-.18408E+01-.13617E+01-.80404E+00-.20313E+00
 .40453E+00 .98351E+00 .15021E+01 .19327E+01 .22543E+01 .24513E+01 .25149E+01
J=  17
 .29806E+01 .28942E+01 .26470E+01 .22551E+01 .17304E+01 .11066E+01 .41015E+00-.30302E+00-.10091E+01-.16590E+01
-.22142E+01-.26370E+01-.29020E+01-.29899E+01-.28953E+01-.26244E+01-.21952E+01-.16350E+01-.98133E+00-.27240E+00
 .44700E+00 .11360E+01 .17571E+01 .22750E+01 .26633E+01 .29023E+01 .29806E+01
J=  18
 .35377E+01 .34350E+01 .31389E+01 .26649E+01 .20382E+01 .12928E+01 .47010E+00-.38335E+00-.12179E+01-.19840E+01
-.26353E+01-.31314E+01-.34412E+01-.35449E+01-.34359E+01-.31211E+01-.26204E+01-.19652E+01-.11960E+01-.35951E+00
 .49462E+00 .13165E+01 .20590E+01 .26825E+01 .31515E+01 .34415E+01 .35377E+01
J=  19
 .42038E+01 .40817E+01 .37261E+01 .31565E+01 .24048E+01 .15131E+01 .55221E+00-.48160E+00-.14694E+01-.23732E+01
-.31392E+01-.37210E+01-.40854E+01-.42081E+01-.40822E+01-.37157E+01-.31305E+01-.23621E+01-.14565E+01-.40740E+00
 .54881E+00 .15273E+01 .24170E+01 .31671E+01 .37330E+01 .40857E+01 .42038E+01
J=  20
 .50000E+01 .48547E+01 .44273E+01 .37420E+01 .28403E+01 .17730E+01 .60208E+00-.60208E+00-.17730E+01-.28403E+01
-.37420E+01-.44273E+01-.48547E+01-.50000E+01-.48547E+01-.44273E+01-.37420E+01-.28403E+01-.17730E+01-.60208E+00
 .60208E+00 .17730E+01 .28403E+01 .37420E+01 .44273E+01 .48547E+01 .50000E+01


*********************************          T-ARRAY          **********************************************
J=   1
-.31000E-04-.14020E-02-.27600E-02-.55520E-02-.85090E-02-.10591E-01-.29900E-01-.60790E-01-.40453E-01-.50524E-01
-.45592E-01-.43734E-01-.16520E-01 .38200E-02 .20435E-01 .48642E-01 .66101E-01 .75249E-01 .74531E-01 .64352E-01
 .47498E-01 .28262E-01 .11299E-01 .68090E-02 .30400E-02 .14260E-02-.31000E-04
J=   2
-.30547E-03-.78795E-02-.16227E-01-.26731E-01-.39785E-01-.57831E-01-.74627E-01-.87647E-01-.94757E-01-.93704E-01
-.82596E-01-.60827E-01-.29950E-01 .62661E-02 .43955E-01 .78021E-01 .10360E+00 .11773E+00 .11952E+00 .10995E+00
 .91541E-01 .67488E-01 .44089E-01 .28150E-01 .16209E-01 .73200E-02-.30547E-03
J=   3
-.90737E-03-.17185E-01-.34510E-01-.53490E-01-.75821E-01-.99488E-01-.11972E+00-.13395E+00-.14024E+00-.13620E+00
-.11920E+00-.87462E-01-.43667E-01 .84442E-02 .61258E-01 .10719E+00 .14081E+00 .15979E+00 .16414E+00 .15540E+00
 .13613E+00 .10469E+00 .80529E-01 .55269E-01 .33904E-01 .15640E-01-.90737E-03
J=   4
-.16095E-02-.28924E-01-.58421E-01-.86120E-01-.11596E+00-.14449E+00-.16746E+00-.18253E+00-.18780E+00-.18050E+00
-.15747E+00-.11620E+00-.58728E-01 .10408E-01 .78892E-01 .13737E+00 .17964E+00 .20391E+00 .21108E+00 .20334E+00
 .18544E+00 .15453E+00 .12057E+00 .86900E-01 .55340E-01 .26093E-01-.16095E-02
J=   5
-.24714E-02-.43448E-01-.84330E-01-.12488E+00-.16306E+00-.19604E+00-.22439E+00-.24046E+00-.24426E+00-.23261E+00
-.20142E+00-.14792E+00-.73931E-01 .12245E-01 .98029E-01 .17120E+00 .22453E+00 .25591E+00 .26697E+00 .26057E+00
 .23976E+00 .20760E+00 .16770E+00 .12470E+00 .81456E-01 .39000E-01-.24714E-02
```

```
J=  6
-.35200E-02-.61791E-01-.11966E+00-.17423E+00-.22510E+00-.26838E+00-.30033E+00-.31818E+00-.31929E+00-.30001E+00
-.25625E+00-.18604E+00-.92671E-01 .14034E-01 .12012E+00 .21206E+00 .28094E+00 .32365E+00 .34139E+00 .33723E+00
 .31459E+00 .27691E+00 .22812E+00 .17278E+00 .11446E+00 .55350E-01-.35200E-02
J=  7
-.47303E-02-.84226E-01-.16162E+00-.23488E+00-.30088E+00-.35561E+00-.39471E+00-.41444E+00-.41125E+00-.38129E+00
-.32[illegible]21E+00-.23066E+00-.11455E+00 .15736E-01 .14529E+00 .25949E+00 .34769E+00 .40573E+00 .43301E+00 .43242E+00
 .40763E+00 .36275E+00 .30241E+00 .23167E+00 .15495E+00 .75[illegible]59E-01-.47303E-02
J=  8
-.60610E-02-.11104E+00-.21259E+00-.30749E+00-.39155E+00-.45977E+00-.50688E+00-.52806E+00-.51895E+00-.47579E+00
-.39635E+00-.28211E+00-.14001E+00 .17323E-01 .17385E+00 .31381E+00 .42518E+00 .50136E+00 .54077E+00 .54526E+00
 .51855E+00 .46541E+00 .39135E+00 .30226E+00 .20356E+00 .99699E-01-.60610E-02
J=  9
-.74554E-02-.14264E+00-.27276E+00-.39314E+00-.49825E+00-.58185E+00-.63762E+00-.65966E+00-.64310E+00-.58444E+00
-.48280E+00-.34157E+00-.16964E+00 .18778E-01 .20635E+00 .37593E+00 .51377E+00 .61134E+00 .66529E+00 .67642E+00
 .64824E+00 .58603E+00 .49623E+00 .38575E+00 .26128E+00 .12659E+00-.74554E-02
J= 10
-.88731E-02-.17964E+00-.34337E+00-.49355E+00-.62300E+00-.72412E+00-.78925E+00-.81168E+00-.78669E+00-.70953E+00
-.58254E+00-.41046E+00-.20424E+00 .20079E-01 .24351E+00 .44722E+00 .61553E+00 .73781E+00 .80887E+00 .82825E+00
 .79906E+00 .72696E+00 .61922E+00 .48399E+00 .32936E+00 .16284E+00-.88731E-02
J= 11
-.10252E-01-.22209E+00-.42610E+00-.61112E+00-.76809E+00-.88997E+00-.96555E+00-.98791E+00-.95164E+00-.85440E+00
-.69828E+00-.49069E+00-.24480E+00 .21195E-01 .28628E+00 .52954E+00 .73306E+00 .88397E+00 .97507E+00 .10045E+01
 .97472E+00 .89165E+00 .76342E+00 .59946E+00 .40964E+00 .20342E+00-.10252E-01
J= 12
-.11513E-01-.27351E+00-.52314E+00-.74910E+00-.93962E+00-.10841E+01-.11714E+01-.11934E+01-.11445E+01-.10233E+01
-.83343E+00-.58464E+00-.29257E+00 .22074E-01 .33570E+00 .62512E+00 .86964E+00 .10539E+01 .11665E+01 .12100E+01
 .11801E+01 .10847E+01 .93288E+00 .73553E+00 .50446E+00 .25157E+00-.11513E-01
J= 13
-.12600E-01-.33284E+00-.63733E+00-.91143E+00-.11409E+01-.13120E+01-.14136E+01-.14343E+01-.13765E+01-.12212E+01
-.99205E+00-.69515E+00-.34903E+00 .22631E-01 .39335E+00 .73670E+00 .10292E+01 .12526E+01 .13949E+01 .14508E+01
 .14212E+01 .13119E+01 .11327E+01 .89627E+00 .61673E+00 .30883E+00-.12600E-01
J= 14
-.13360E-01-.40267E+00-.77201E+00-.11031E+01-.13781E+01-.15810E+01-.16976E+01-.17179E+01-.16366E+01-.14544E+01
-.11791E+01-.82566E+00-.41601E+00 .22751E-01 .46058E+00 .86748E+00 .12164E+01 .14859E+01 .16610E+01 .17343E+01
 .17055E+01 .15801E+01 .13690E+01 .10860E+01 .75008E+00 .37714E+00-.13360E-01
J= 15
-.13700E-01-.48500E+00-.93144E+00-.13300E+01-.16591E+01-.18992E+01-.20343E+01-.20533E+01-.19511E+01-.17300E+01
-.14003E+01-.98027E+00-.49569E+00 .22273E-01 .53933E+00 .10212E+01 .14369E+01 .17609E+01 .19750E+01 .20692E+01
 .20417E+01 .18979E+01 .16495E+01 .13132E+01 .90691E+00 .45803E+00-.13700E-01
J= 16
-.13406E-01-.58244E+00-.11207E+01-.15996E+01-.19928E+01-.22770E+01-.24338E+01-.24509E+01-.23239E+01-.20567E+01
-.16626E+01-.11639E+01-.59067E+00 .20969E-01 .63176E+00 .12025E+01 .16972E+01 .20858E+01 .23464E+01 .24659E+01
 .24406E+01 .22754E+01 .19433E+01 .15831E+01 .10986E+01 .55680E+00-.13406E-01
J= 17
-.12292E-01-.69804E+00-.13460E+01-.19208E+01-.23902E+01-.27267E+01-.29090E+01-.29237E+01-.27669E+01-.24448E+01
-.19743E+01-.13823E+01-.70409E+00 .18548E-01 .74043E+00 .14165E+01 .20051E+01 .24707E+01 .27869E+01 .29370E+01
 .29149E+01 .27252E+01 .23815E+01 .19056E+01 .13256E+01 .67457E+00-.12292E-01
J= 18
-.99564E-02-.83561E+00-.16148E+01-.23043E+01-.28648E+01-.32634E+01-.34756E+01-.34870E+01-.32943E+01-.29068E+01
-.23455E+01-.16426E+01-.83972E+00 .14598E-01 .86838E+00 .16696E+01 .23697E+01 .29273E+01 .33102E+01 .34975E+01
 .34803E+01 .32621E+01 .28577E+01 .22921E+01 .15985E+01 .81661E+00-.99564E-02
J= 19
-.60318E-02-.99962E+00-.19368E+01-.27637E+01-.34329E+01-.39053E+01-.41527E+01-.41593E+01-.39235E+01-.34575E+01
-.27878E+01-.19530E+01-.10021E+01 .86308E-02 .10190E+01 .19690E+01 .28022E+01 .34698E+01 .39330E+01 .41657E+01
 .41555E+01 .39046E+01 .34287E+01 .27563E+01 .19270E+01 .98833E+00-.60318E-02
J= 20
0.          -.11966E+01-.23236E+01-.33156E+01-.41149E+01-.46751E+01-.49635E+01-.49635E+01-.46751E+01-.41149E+01
-.33156E+01-.23236E+01-.11966E+01 .58559E-12 .11966E+01 .23236E+01 .33156E+01 .41149E+01 .46751E+01 .49635E+01
 .49635E+01 .46751E+01 .41149E+01 .33156E+01 .23236E+01 .11966E+010.
```

Sample Case Plot: Single-Body Field





Sample Case Input: Double-Body Field

```
TEST CASE - BODY-FITTED COORDINATE SYSTEM
DOUBLE BODY : TWO KARMAN-TREFFTZ AIRFOILS, 26 POINTS ON EACH
K-T AIRFOIL-FLAP #1
   69   20    2  200    0    1    1    0    0
    1    3    1    0    0   10    0    1    1
    4    2
    1    1   14   -1
    1   56   69   -1
    1   22   48   -2
    3    1   69    0
    1   14   22    1   48   56
    2    1   20    4    1   20
1.8        0.0001     0.0001     5.0        0.0        0.0        0.0             68
    0    0 0.0
    0    0
8.0        0.0        10.0       0.0        0.0        0.0        0.0
   .808147  -.162448
   .807780  -.162631
   .807061  -.162608
   .804591  -.161989
   .800795  -.160650
   .782146  -.152794
   .754614  -.140182
   .721468  -.124170
   .686409  -.106140
   .653046  -.087505
   .624771  -.069728
   .604009  -.054319
   .592603  -.042765
   .591634  -.036332
   .601308  -.035389
   .620148  -.039856
   .645928  -.049700
   .676123  -.064478
   .708095  -.083158
   .739231  -.104073
   .767046  -.124998
   .789221  -.143353
   .803601  -.156552
   .806312  -.159414
   .807858  -.161394
   .808167  -.162039
   .499528  -.000031
   .498623  -.001402
   .496086  -.002760
   .486293  -.005552
   .470464  -.008509
   .390148  -.018591
   .269552  -.029966
   .122707  -.040790
  -.034802  -.048453
  -.187472  -.050524
  -.321146  -.045592
  -.423886  -.033739
  -.486506  -.016526
  -.502726   .003820
  -.471102   .026435
  -.396903   .048642
  -.287913   .066101
  -.153756   .075299
```

```
   -.005644    .074531
    .144046    .064352
    .282251    .047498
    .395776    .028262
    .471990    .011299
    .487103    .006809
    .496419    .003040
    .498774    .001426
ETA         0      4     2 0.0         10.0
            1 100.0     1.0
            2 100.0     1.0
            3 100.0     1.0
            4 100.0     1.0
    22      1 100.0     1.0
    48      1 100.0     1.0
XI          0      0     0 0.0         0.0
     3      0 100.0     1.0
```

Sample Case Output: Double-Body Field

```
***************** INPUT *****************************

IMAX,JMAX,NBDY,ITER,IGES,IDISK,IRIN,IRINTL,INPIN,IGEO :   69   20    4  200    0    1    1    0    0    0

IPLOT,IPLTR,NCOPY,LINRT1,LINRT2,NUMBR,NUMBR1,ISKIP1,ISKIP2 :    1    5    1    0    0   10    0    1    1

NBSEG,NRSEG :    4    2

R(1),R(2),R(3),TINFIN,AINFIN,XOINF,YOINF,NINF :
    1.80000000     .00010000     .00010000     5.00000000     0.00000000     0.00000000     0.00000000   68

IEV,IAIT,R(10) :    0    0     0.00000000

INFAC,INFACD :    0    0

SIZE,RATIO,DIST,XB1,XB2,YB1,YB2  0.00000000  0.00000000 10.00000000  0.00000000  0.00000000  0.00000000  0.00000000
*****************************************************************************************************************

INITIAL GUESS TYPE:   0

**BODY SEGMENTS**

  L  LBSID    LB1    LB2   LBDY

  1      1      1     14     -1
  2      1     56     69     -1
  3      1     22     48     -2
  4      3      1     69      0

**REENTRANT SEGMENTS**

  L  LRSID    LR1    LR2  LISID    LI1    LI2

  1      1     14     22      1     48     56
  2      2      1     20      4      1     20

**OUTER BOUNDARY**

  RADIUS =  5.00000000          INITIAL ANGLE =    0.00000000
  ORIGIN AT X = 0.00000000  , Y = 0.00000000

  NUMBER OF POINTS =  68              1 STEPS IN ATTAINMENT OF INFINITY          INITIAL STEP (LINEAR CASE) =   1

UNIFORM ACCELERATION PARAMETER USED.
```

```
TEST CASE - BOUNDARY-FITTED COORDINATE SYSTEM
DOUBLE BODY - TWO KARMAN-TREFFTZ AIRFOILS, 20 POINTS ON EACH

                    BOUNDARY-FITTED COORDINATE SYSTEM

TRANSFORMED BODY, KT AIRFOIL-FLAP #1

FIELD PARAMETERS, NUMBER OF XI-LINES =  60
                  NUMBER OF ETA-LINES =   20

ITERATION PARAMETERS: SOR ACCELERATION PARAMETER =  1.80000
                      MAXIMUM NUMBER OF ITERATIONS ALLOWED =  200
                      ALLOWABLE ITERATION ERROR NORMS: X: .10000E-03
                                                       Y: .10000E-03

NUMBER OF BODIES IN FIELD =    2

PLOT PARAMETERS:  COPIES DESIRED =   1
                  LINEWEIGHT DESIRED =   6
                  PLOT SIZE IN Y-DIRECTION = 8.000
                  RATIO = 0.000

LOGICAL CONTROLS : LACC = VARIABLE ACCELERATION PARAMETER FIELD CONVERGED (IRRELEVANT IF LCON=TRUE)

                   LCFAC = ADDITION OF INHOMOGENEOUS TERM COMPLETED
                   LCON = UNIFORM ACCELERATION PARAMETER


EXPONENTIAL DECAY RHS
* ATTRACTION *

*** ETA EQUATION RHS ***

ATTRACTION LINES

   J            AMP                 DECAY
   1       1000.00000000          1.00000000
   2       1000.00000000          1.00000000
   3       1000.00000000          1.00000000
   4       1000.00000000          1.00000000

ATTRACTION POINTS

   I    J            AMP                 DECAY    LPT
  20    1       1000.00000000          1.00000000    0
  40    1       1000.00000000          1.00000000    0

    5 STEPS IN ADDITION OF INHOMOGENEOUS TERM,   INTERMEDIATE CONVERGENCE FACTOR =   100.00000000

      .20000000 OF INHOMOGENEOUS TERM

     1.00000000 OF INNER BOUNDARY

***** MAXIMUM NORMS OF ITERATE CHANGES                                    LOGICAL CONTROLS


ITERATE        XNORM          YNORM        ALLOWNORM        AVG.ACC.PARA.     LACC      LCFAC       LCON
      1     .14902E+01     .14890E+00    0.                     1.80000       T         F           T
      2     .90017E+00     .15291E+00    0.                     1.80000       T         F           T
      3     .21410E+01     .18082E+01    0.                     1.80000       T         F           T
      4     .91213E+00     .24250E+00    0.                     1.80000       T         F           T
      5     .71030E+00     .15903E+00    0.                     1.80000       T         F           T
      6     .83747E+00     .17998E+00    0.                     1.80000       T         F           T
      7     .55800E+00     .14717E+00    0.                     1.80000       T         F           T
      8     .40253E+00     .19882E+00    0.                     1.80000       T         F           T
      9     .43770E+00     .27010E+00    0.                     1.80000       T         F           T
```

10	.56745E+00	.20083E+00	0.	1.00000	T	*	T
11	.67220E+00	.21043E+00	0.	1.00000	T	*	T
12	.70723E+00	.21483E+00	0.	1.00000	T	*	T
13	.62115E+00	.21791E+00	0.	1.00000	T	*	T
14	.70704E+00	.21512E+00	0.	1.00000	T	*	T
15	.66406E+00	.20496E+00	0.	1.00000	T	*	T
16	.54205E+00	.104[illegible]E+00	0.	1.00000	T	*	T
17	.46740E+00	.18074E+00	0.	1.00000	T	*	T
18	.38245E+00	.17427E+00	0.	1.00000	T	*	T
19	.34247E+00	.16672E+00	0.	1.00000	T	*	T
20	.25359E+00	.15251E+00	0.	1.00000	T	*	T
21	.20111E+00	.14708E+00	0.	1.00000	T	*	T
22	.19005E+00	.12427E+00	0.	1.00000	T	*	T
23	.16223E+00	.13271E+00	0.	1.00000	T	*	T
24	.16527E+00	.13411E+00	0.	1.00000	T	*	T
25	.15649E+00	.11720E+00	0.	1.00000	T	*	T
26	.13645E+00	.12255E+00	0.	1.00000	T	*	T
27	.14057E+00	.92025E-01	0.	1.00000	T	*	T
28	.13700E+00	.86375E-01	0.	1.00000	T	*	T
29	.12530E+00	.79664E-01	0.	1.00000	T	*	T
30	.15755E+00	.73502E-01	0.	1.00000	T	*	T
31	.11004E+00	.61300E-01	0.	1.00000	T	*	T
32	.11220E+00	.56330E-01	0.	1.00000	T	*	T
33	.49730E-01	.55532E-01	0.	1.00000	T	*	T
34	.69610E-01	.47543E-01	0.	1.00000	T	*	T
35	.87387E-01	.48695E-01	0.	1.00000	T	*	T
36	.94404E-01	.49763E-01	0.	1.00000	T	*	T
37	.85117E-01	.51295E-01	0.	1.00000	T	*	T
38	.76940E-01	.55242E-01	0.	1.00000	T	*	T
39	.72599E-01	.52747E-01	0.	1.00000	T	*	T
40	.67705E-01	.50591E-01	0.	1.00000	T	*	T
41	.63202E-01	.54367E-01	0.	1.00000	T	*	T
42	.55535E-01	.54142E-01	0.	1.00000	T	*	T
43	.51787E-01	.46140E-01	0.	1.00000	T	*	T
44	.44050E-01	.34603E-01	0.	1.00000	T	*	T
45	.34752E-01	.35344E-01	0.	1.00000	T	*	T
46	.36651E-01	.31057E-01	0.	1.00000	T	*	T
47	.36632E-01	.27620E-01	0.	1.00000	T	*	T
48	.40231E-01	.25495E-01	0.	1.00000	T	*	T
49	.34298E-01	.23062E-01	0.	1.00000	T	*	T
50	.29100E-01	.20650E-01	0.	1.00000	T	*	T
51	.30427E-01	.19346E-01	0.	1.00000	T	*	T
52	.29114E-01	.17547E-01	0.	1.00000	T	*	T
53	.25569E-01	.16175E-01	0.	1.00000	T	*	T
54	.23321E-01	.14068E-01	0.	1.00000	T	*	T
55	.18202E-01	.14109E-01	0.	1.00000	T	*	T
56	.12549E-01	.13620E-01	0.	1.00000	T	*	T
57	.89149E-02	.13012E-01	0.	1.00000	T	*	T
58	.65005E-02	.11081E-01	0.	1.00000	T	*	T
59	.43774E-02	.98312E-02	0.	1.00000	T	*	T

.50000000 OF INHOMOGENEOUS TERM

60	.35640E-01	.34850E-01	0.	1.00000	T	*	T
61	.29573E-01	.31435E-01	0.	1.00000	T	*	T
62	.23192E-01	.27059E-01	0.	1.00000	T	*	T
63	.17454E-01	.24073E-01	0.	1.00000	T	*	T
64	.14560E-01	.20251E-01	0.	1.00000	T	*	T
65	.22646E-01	.26652E-01	0.	1.00000	T	*	T
66	.21609E-01	.31149E-01	0.	1.00000	T	*	T
67	.15363E-01	.25230E-01	0.	1.00000	T	*	T
68	.10464E-01	.14650E-01	0.	1.00000	T	*	T
69	.10535E-01	.13660E-01	0.	1.00000	T	*	T
70	.10375E-01	.13064E-01	0.	1.00000	T	*	T
71	.10170E-01	.11010E-01	0.	1.00000	T	*	T
72	.10085E-01	.99035E-02	0.	1.00000	T	*	T
73	.98030E-02	.93275E-02	0.	1.00000	T	*	T

1.0000000 OF [illegible]

74	.47991E-01	.34262E-01	0.	1.00000	T	T	T
75	.31000E-01	.33317E-01	0.	1.00000	T	T	T
76	.24422E-01	.24449E-01	0.	1.00000	T	T	T
77	.16734E-01	.25142E-01	0.	1.00000	T	T	T
78	.16227E-01	.21044E-01	0.	1.00000	T	T	T
79	.24901E-01	[illegible]	0.	1.00000	T	T	T
80	[illegible]	.34714E-01	0.	1.00000	T	T	T
81	.30407E-01	[illegible]	0.	1.00000	T	T	T
82	.21331E-01	[illegible]	0.	1.00000	T	T	T
83	.15449E-01	.21049E-01	0.	1.00000	T	T	T
84	[illegible]	.16712E-01	0.	1.00000	T	T	T
85	.11244E-01	.14834E-01	0.	1.00000	T	T	T
86	.11006E-01	.12414E-01	0.	1.00000	T	T	T
87	.10737E-01	.11424E-01	0.	1.00000	T	T	T
88	.10384E-01	.11010E-01	0.	1.00000	T	T	T
89	[illegible]	.47724E-02	0.	1.00000	T	T	T
90	[illegible]	[illegible]	0.	1.00000	T	T	T
91	.73013E-02	.83171E-02	0.	1.00000	T	T	T
92	.69420E-02	.77443E-02	0.	1.00000	T	T	T
93	.66837E-02	.73170E-02	0.	1.00000	T	T	T
94	.67301E-02	.66421E-02	0.	1.00000	T	T	T
95	.63603E-02	.60473E-02	0.	1.00000	T	T	T
96	.54019E-02	.54504E-02	0.	1.00000	T	T	T
97	.53451E-02	.44604E-02	0.	1.00000	T	T	T
98	.45958E-02	.43204E-02	0.	1.00000	T	T	T
99	.34370E-02	.34454E-02	0.	1.00000	T	T	T
100	.27126E-02	.33526E-02	0.	1.00000	T	T	T
101	.26253E-02	.32771E-02	0.	1.00000	T	T	T
102	.25570E-02	.30054E-02	0.	1.00000	T	T	T
103	.24042E-02	.27159E-02	0.	1.00000	T	T	T
104	.23234E-02	.23445E-02	0.	1.00000	T	T	T
105	.22070E-02	.21044E-02	0.	1.00000	T	T	T
106	.20557E-02	.14049E-02	0.	1.00000	T	T	T
107	.16653E-02	.15144E-02	0.	1.00000	T	T	T
108	.16434E-02	.12770E-02	0.	1.00000	T	T	T
109	.10442E-02	.11430E-02	0.	1.00000	T	T	T
110	.12427E-02	.10630E-02	0.	1.00000	T	T	T
111	.11437E-02	.99514E-03	0.	1.00000	T	T	T
112	.10557E-02	.94203E-03	0.	1.00000	T	T	T
113	.94624E-03	.84127E-03	0.	1.00000	T	T	T
114	.82915E-03	.87024E-03	0.	1.00000	T	T	T
115	.67704E-03	.86540E-03	0.	1.00000	T	T	T
116	.67501E-03	.84045E-03	0.	1.00000	T	T	T
117	.63342E-03	.61940E-03	0.	1.00000	T	T	T
118	.62304E-03	.77500E-03	0.	1.00000	T	T	T
119	.67541E-03	.71377E-03	0.	1.00000	T	T	T
120	.72383E-03	.62344E-03	0.	1.00000	T	T	T
121	.73280E-03	.51755E-03	0.	1.00000	T	T	T
122	.72641E-03	.40479E-03	0.	1.00000	T	T	T
123	.60713E-03	.42344E-03	0.	1.00000	T	T	T
124	.61400E-03	.41217E-03	0.	1.00000	T	T	T
125	.53472E-03	.39413E-03	0.	1.00000	T	T	T
126	.49150E-03	.34810E-03	0.	1.00000	T	T	T
127	.41925E-03	.37447E-03	0.	1.00000	T	T	T
128	.33263E-03	.31004E-03	0.	1.00000	T	T	T
129	.26527E-03	.20345E-03	0.	1.00000	T	T	T
130	.21240E-03	.17402E-03	0.	1.00000	T	T	T
131	.17224E-03	.15655E-03	0.	1.00000	T	T	T
132	.14370E-03	.15431E-03	0.	1.00000	T	T	T
133	.13194E-03	.10247E-03	0.	1.00000	T	T	T
134	.11705E-03	.17221E-03	0.	1.00000	T	T	T
135	.96708E-04	.16744E-03	0.	1.00000	T	T	T
136	.74424E-04	.15115E-03	0.	1.00000	T	T	T
137	.65777E-04	.12831E-03	0.	1.00000	T	T	T
138	.51950E-04	.10440E-03	0.	1.00000	T	T	T
139	.47679E-04	[illegible]	0.	1.00000	T	T	T

[illegible]

FINAL VALUES

ITERATION CONVERGED.

INITIAL ITERATION ERROR NORMS XI [illegible] YI [illegible] AT ITERATE = 1
FINAL ITERATION ERROR NORMS XI [illegible] YI [illegible] AT ITERATE = [illegible]
LOCATION OF MAXIMUM ITERATION ERRORS XI I= [illegible] J= 15
YI I= 21 J= 10

J= 1

[illegible]

J= 2

[illegible]

J= 3

[illegible]

J= 4

[illegible]

J= 5

[illegible]

J= 6

[illegible]

J= 7

[illegible]

J= 8

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

J= 9

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

J= 10

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

J= 11

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

J= 12

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

J= 13

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

J= 14

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

J= 15

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

J= 16

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

J= 17

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

J= 18

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

J= 19

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

J= 20

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

****************************** [illegible] ******************************

J= 1

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

J= 2

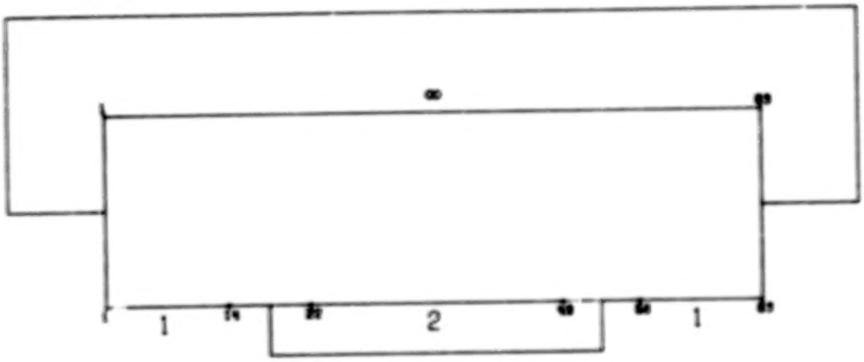
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

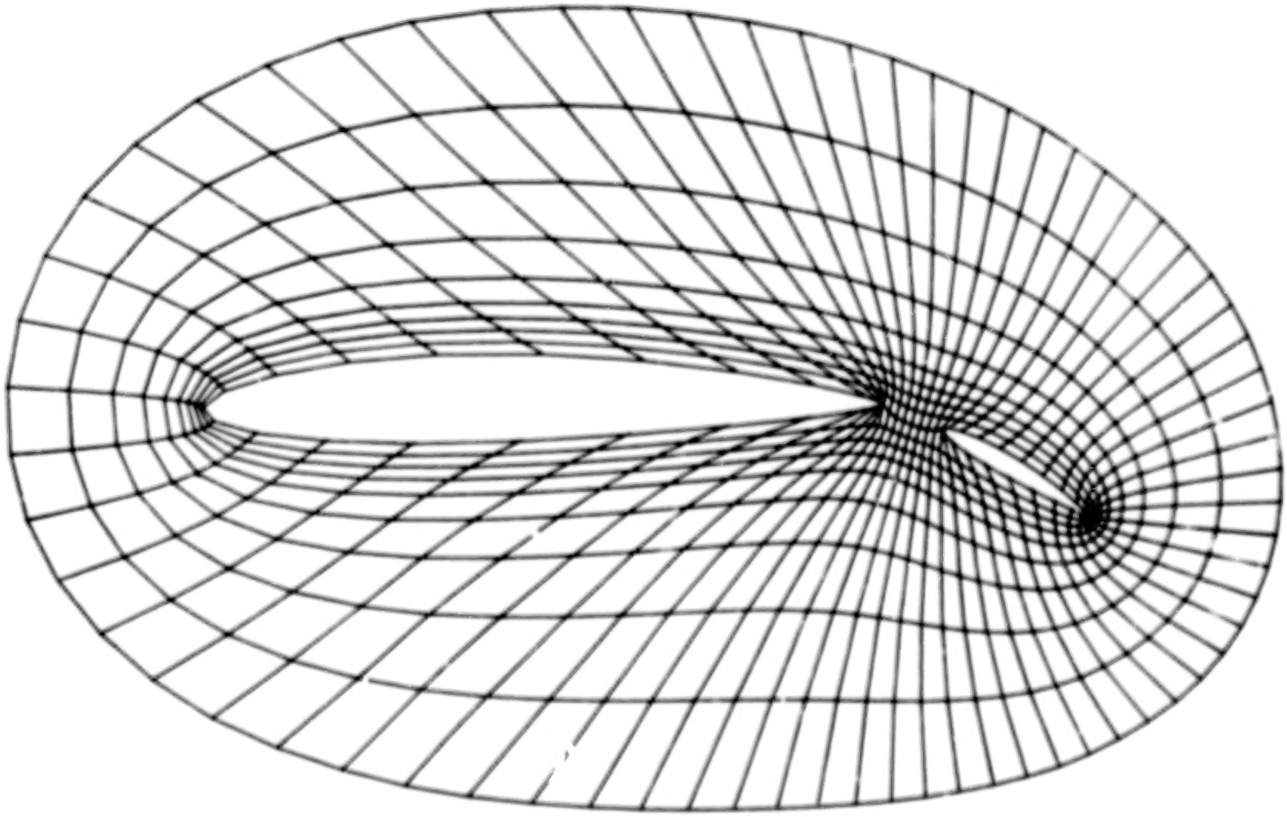
```
J=  3
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
J=  4
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
J=  5
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
J=  6
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
J=  7
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
J=  8
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
J=  9
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
J= 10
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
```

```
J=  11
-.20153E+00-.25475E+00-.31598E+00-.36970E+00-.42048E+00-.46795E+00-.51170E+00-.55179E+00-.58702E+00-.61991E+00
-.64608E+00-.67253E+00-.69342E+00-.71107E+00-.72584E+00-.73745E+00-.74600E+00-.75121E+00-.75732E+00-.75870E+00
-.75700E+00-.75104E+00-.74149E+00-.72849E+00-.70521E+00-.67565E+00-.63085E+00-.58765E+00-.52721E+00-.45520E+00
-.37200E+00-.27873E+00-.17724E+00-.70032E-01 .39970E-01 .14901E+00 .25580E+00 .35504E+00 .44527E+00 .52419E+00
 .59053E+00 .64370E+00 .68411E+00 .71220E+00 .72430E+00 .73678E+00 .73584E+00 .72784E+00 .71388E+00 .69492E+00
 .67170E+00 .64470E+00 .61448E+00 .58113E+00 .54488E+00 .50580E+00 .46395E+00 .41935E+00 .37208E+00 .32213E+00
 .26970E+00 .21496E+00 .15810E+00 .99712E-01 .39964E-01-.20573E-01-.81383E-01-.14189E+00-.20153E+00
J=  12
-.21117E+00-.28554E+00-.35782E+00-.42742E+00-.49384E+00-.55660E+00-.61538E+00-.66982E+00-.71979E+00-.76504E+00
-.80578E+00-.84170E+00-.87328E+00-.90020E+00-.92291E+00-.94122E+00-.95525E+00-.96487E+00-.96992E+00-.97084E+00
-.96674E+00-.95335E+00-.93506E+00-.90890E+00-.87386E+00-.82890E+00-.77312E+00-.70584E+00-.62677E+00-.53609E+00
-.43454E+00-.32345E+00-.20464E+00-.80629E-01 .46035E-01 .17239E+00 .29549E+00 .41251E+00 .52090E+00 .61857E+00
 .70395E+00 .77660E+00 .83452E+00 .87948E+00 .91152E+00 .93151E+00 .94053E+00 .93970E+00 .93013E+00 .91285E+00
 .88874E+00 .85860E+00 .82301E+00 .78251E+00 .73740E+00 .68819E+00 .63494E+00 .57795E+00 .51740E+00 .45352E+00
 .38650E+00 .31660E+00 .24473E+00 .17061E+00 .94993E-01 .18401E-01-.58901E-01-.13530E+00-.21117E+00
J=  13
-.21992E+00-.31453E+00-.40483E+00-.49021E+00-.58199E+00-.66308E+00-.74074E+00-.81282E+00-.87952E+00-.94088E+00
-.99603E+00-.10455E+01-.10890E+01-.11265E+01-.11578E+01-.11828E+01-.12014E+01-.12134E+01-.12183E+01-.12157E+01
-.12051E+01-.11856E+01-.11567E+01-.11174E+01-.10670E+01-.10048E+01-.93022E+00-.84308E+00-.74352E+00-.63208E+00
-.50960E+00-.37616E+00-.23912E+00-.94985E-01 .51052E-01 .19802E+00 .34132E+00 .47882E+00 .60803E+00 .72676E+00
 .83324E+00 .92614E+00 .10048E+01 .10687E+01 .11180E+01 .11532E+01 .11749E+01 .11839E+01 .11814E+01 .11681E+01
 .11452E+01 .11134E+01 .10735E+01 .10263E+01 .97238E+00 .91228E+00 .84648E+00 .77541E+00 .69950E+00 .61914E+00
 .53478E+00 .44687E+00 .35541E+00 .26248E+00 .16710E+00 .70492E-01-.26695E-01-.12373E+00-.21992E+00
J=  14
-.22812E+00-.34580E+00-.46316E+00-.57697E+00-.68670E+00-.79173E+00-.89127E+00-.98478E+00-.10718E+01-.11519E+01
-.12248E+01-.12900E+01-.13474E+01-.13967E+01-.14378E+01-.14698E+01-.14929E+01-.15067E+01-.15105E+01-.15039E+01
-.14863E+01-.14570E+01-.14154E+01-.13609E+01-.12929E+01-.12111E+01-.11154E+01-.10059E+01-.88311E+00-.74792E+00
-.60161E+00-.44525E+00-.28270E+00-.11448E+00 .56202E-01 .22684E+00 .39400E+00 .55582E+00 .70934E+00 .85238E+00
 .98293E+00 .10995E+01 .12009E+01 .12864E+01 .13550E+01 .14080E+01 .14450E+01 .14673E+01 .14743E+01 .14675E+01
 .14477E+01 .14159E+01 .13729E+01 .13197E+01 .12570E+01 .11855E+01 .11061E+01 .10193E+01 .92577E+00 .82621E+00
 .72124E+00 .61154E+00 .49781E+00 .38075E+00 .26110E+00 .13983E+00 .17604E-01-.10468E+00-.22812E+00
J=  15
-.22751E+00-.37644E+00-.52652E+00-.67077E+00-.81020E+00-.94391E+00-.10710E+01-.11908E+01-.13026E+01-.14058E+01
-.14994E+01-.15834E+01-.16572E+01-.17201E+01-.17719E+01-.18120E+01-.18399E+01-.18551E+01-.18571E+01-.18452E+01
-.18190E+01-.17774E+01-.17214E+01-.16492E+01-.15610E+01-.14568E+01-.13368E+01-.12015E+01-.10519E+01-.88895E+00
-.71438E+00-.52947E+00-.33747E+00-.14081E+00 .56875E-01 .25832E+00 .45475E+00 .64542E+00 .82773E+00 .99928E+00
 .11579E+01 .13018E+01 .14294E+01 .15400E+01 .16323E+01 .17063E+01 .17617E+01 .17988E+01 .18182E+01 .18203E+01
 .18059E+01 .17756E+01 .17311E+01 .16720E+01 .16012E+01 .15179E+01 .14236E+01 .13193E+01 .12057E+01 .10839E+01
 .95470E+00 .81918E+00 .67878E+00 .53248E+00 .38332E+00 .23169E+00 .78629E-01-.74790E-01-.22751E+00
J=  16
-.22139E+00-.41355E+00-.59685E+00-.77868E+00-.95479E+00-.11240E+01-.12851E+01-.14372E+01-.15792E+01-.17102E+01
-.18294E+01-.19360E+01-.20292E+01-.21084E+01-.21729E+01-.22220E+01-.22551E+01-.22717E+01-.22710E+01-.22527E+01
-.22161E+01-.21610E+01-.20870E+01-.19941E+01-.18823E+01-.17519E+01-.16036E+01-.14382E+01-.12570E+01-.10613E+01
-.85315E+00-.63450E+00-.40794E+00-.17600E+00 .58842E-01 .29300E+00 .52438E+00 .74975E+00 .96642E+00 .11718E+01
 .13630E+01 .15347E+01 .16948E+01 .18379E+01 .19575E+01 .20563E+01 .21338E+01 .21899E+01 .22247E+01 .22385E+01
 .22320E+01 .22057E+01 .21607E+01 .20977E+01 .20178E+01 .19222E+01 .18119E+01 .16880E+01 .15517E+01 .14042E+01
 .12467E+01 .10805E+01 .90683E+00 .72655E+00 .54153E+00 .35290E+00 .16201E+00-.29771E-01-.22109E+00
J=  17
-.22062E+00-.43975E+00-.67337E+00-.90178E+00-.11234E+01-.13365E+01-.15399E+01-.17317E+01-.19110E+01-.20763E+01
-.22265E+01-.23605E+01-.24774E+01-.25760E+01-.26558E+01-.27154E+01-.27545E+01-.27723E+01-.27683E+01-.27420E+01
-.26930E+01-.26211E+01-.25264E+01-.24090E+01-.22694E+01-.21082E+01-.19265E+01-.17255E+01-.15068E+01-.12722E+01
-.10240E+01-.76452E+00-.49641E+00-.22251E+00 .54245E-01 .33081E+00 .60413E+00 .87119E+00 .11290E+01 .13749E+01
 .16062E+01 .18205E+01 .20157E+01 .21901E+01 .23421E+01 .24706E+01 .25747E+01 .26540E+01 .27081E+01 .27371E+01
 .27413E+01 .27213E+01 .26777E+01 .26114E+01 .25234E+01 .24150E+01 .22873E+01 .21417E+01 .19796E+01 .18024E+01
 .16118E+01 .14092E+01 .11963E+01 .97478E+00 .74626E+00 .51252E+00 .27528E+00 .36320E-01-.22062E+00
```

```
J=  18
-.16610E+00-.46226E+00-.75466E+00-.10410E+01-.13191E+01-.15870E+01-.18425E+01-.20839E+01-.23093E+01-.25171E+01
-.27055E+01-.28732E+01-.30186E+01-.31412E+01-.32390E+01-.33114E+01-.33576E+01-.33768E+01-.33685E+01-.33324E+01
-.32683E+01-.31763E+01-.30567E+01-.29101E+01-.27373E+01-.25395E+01-.23180E+01-.20745E+01-.18112E+01-.15302E+01
-.12340E+01-.92549E+00-.60754E+00-.28325E+00 .44180E-01 .37147E+00 .69532E+00 .10125E+01 .13198E+01 .16141E+01
 .18926E+01 .21526E+01 .23917E+01 .26077E+01 .27988E+01 .29630E+01 .30995E+01 .32072E+01 .32855E+01 .33340E+01
 .33527E+01 .33419E+01 .33021E+01 .32341E+01 .31387E+01 .30171E+01 .28767E+01 .27010E+01 .25097E+01 .22984E+01
 .20690E+01 .18236E+01 .15641E+01 .12927E+01 .10115E+01 .72273E+00 .42869E+00 .13166E+00-.16610E+00
J=  19
-.10285E+00-.47243E+00-.83611E+00-.11968E+01-.15455E+01-.18817E+01-.22028E+01-.25060E+01-.27889E+01-.30494E+01
-.32853E+01-.34946E+01-.36756E+01-.38268E+01-.39468E+01-.40346E+01-.40888E+01-.41092E+01-.40954E+01-.40471E+01
-.39645E+01-.38480E+01-.36985E+01-.35165E+01-.33038E+01-.30619E+01-.27929E+01-.24988E+01-.21818E+01-.18451E+01
-.14915E+01-.11242E+01-.74640E+00-.36159E+00 .26729E-01 .41497E+00 .79954E+00 .11768E+01 .15435E+01 .18961E+01
 .22314E+01 .25463E+01 .28381E+01 .31041E+01 .33421E+01 .35500E+01 .37261E+01 .38689E+01 .39775E+01 .40510E+01
 .40891E+01 .40918E+01 .40588E+01 .39912E+01 .38896E+01 .37550E+01 .35889E+01 .33927E+01 .31684E+01 .29179E+01
 .26435E+01 .23476E+01 .20326E+01 .17012E+01 .13561E+01 .10002E+01 .63647E+00 .26779E+00-.10285E+00
J=  20
0.         -.46134E+00-.91875E+00-.13683E+01-.18062E+01-.22287E+01-.26322E+01-.30132E+01-.33685E+01-.36950E+01
-.39901E+01-.42511E+01-.44758E+01-.46624E+01-.48091E+01-.49149E+01-.49787E+01-.50000E+01-.49787E+01-.49149E+01
-.48091E+01-.46624E+01-.44758E+01-.42511E+01-.39901E+01-.36950E+01-.33685E+01-.30132E+01-.26322E+01-.22287E+01
-.18062E+01-.13683E+01-.91875E+00-.46134E+00 .58559E-12 .46134E+00 .91875E+00 .13683E+01 .18062E+01 .22287E+01
 .26322E+01 .30132E+01 .33685E+01 .36950E+01 .39901E+01 .42511E+01 .44758E+01 .46624E+01 .48091E+01 .49149E+01
 .49787E+01 .50000E+01 .49787E+01 .49149E+01 .48091E+01 .46624E+01 .44758E+01 .42511E+01 .39901E+01 .36950E+01
 .33685E+01 .30132E+01 .26322E+01 .22287E+01 .18062E+01 .13683E+01 .91875E+00 .46134E+000,
```

Sample Case Plot: Double-Body Field





VII. INSTRUCTIONS FOR USE - SCALE FACTORS

After a coordinate system has been generated, the "scale factors" for use in the solution in any partial differential equation transformed to the rectangular transformed plane are generated and written on a disk file by the main program FATCAT with its subroutine ABC, AMXMN, PARA1, PARA2, PARA3, and WRDATA. Instructions for use are included in the listing of FATCAT.

Core must be set to zero at load time.

Dimensions. The standard program allows a maximum field size of 70 ξ lines and 60 η lines and requires a core size of 201,000 words for the Langley Research Center's CDC 6000 Series Computer System.

Input Parameters. Explanation found with the sample test data, and program listing.

Files. This program requires 2 essential files:

TAPE 1 - input tape - generated as TAPE10 by Program TOMCAT.

TAPE10 - disk on which the factors are to be written.

Scale Factors

Program FATCAT

```
      PROGRAM FATCAT(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
     1TAPE10,TAPE1)
C   ********** MISSISSIPPI STATE 2-D COORDINATE TRANSFORMATION **********
C   *
C   * COORDINATE SYSTEM 'SCALE FACTORS': ALPHA,BETA,GAMA,SIGMA,TAU,
C   *                                    JACOBIAN,DX/DXI,DX/DETA,
C   *                                    DY/DXI, AND DY/DETA
C   *
C   ************************************************************************
C
      DIMENSION X(70,60), Y(70,60), C(10,70,60), CB(10,70,4)
      DIMENSION C1(8),C2(8),LBSID(6),LB1(6),LB2(6),LBDY(6),         LRSID
     1(6), LR1(6), LR2(6), LISID(6), LI1(6), LI2(6), LTYPE(6), LBEN(6)
      DATA NDIM,NDIMY,NDIMX /70,60,70/
C*****
C
C   ************************************************************************
C   *
C   *
C   *      THE COORDINATE SYSTEM IS READ FROM DISK(TAPE1) IN THE SAME
C   *      FORMAT USED BY TOMCAT TO WRITE ON DISK.  THE SCALE FACTORS
C   *      ARE WRITTEN ON DISK(TAPE10) IN ONE OF THE FOLLOWING FORMATS,
C   *      SPECIFIED BY IFORM.
C   *
C   *                *** FORMAT #1 (IFORM=1) ***
C   *
C   *      WRITE(10,1) C1
C   *      WRITE(10,1) C1
C   *      WRITE(10,1) IMAX,JMAX,NBSEG,NRSEG,LISEG,NBDY
C   *      WRITE(10,1) (LBSID(L),LB1(L),LB2(L),LBDY(L),LBEN(L),
C   *     1 L=1,NBSEG)
C   *      WRITE(10,1) (LRSID(L),LR1(L),LR2(L),LISID(L),LI1(L),LI2(L),
C   *     1 LTYPE(L),L=1,NRSEG)
C   *      DO 2 J=1,JMAX-2
C   *    2 WRITE(10,1) ((C(N,I,J),N=1,10),I=1,IMAX-2)
C   *      DO 3 J=1,4
C   *    3 WRITE(10,1) ((CB(N,I,J),N=1,10),I=1,MAX0(IMAX,JMAX)
C   *
C   *      THE ARRAY C CONTAINS THE FACTORS FROM THE SECOND POINT TO THE
C   *      PENULTIMATE POINT, ON SECOND ROW TO THE PENULTIMATE ROW.
C   *      THE ARRAY CB CONTAINS THE FACTORS ON THE RECTANGULAR BOUNDARY,
C   *      J=1,2,3,4 IN CB CORRESPONDING, RESPECTIVELY, TO THE BOTTOM,
C   *      LEFT, TOP, RIGHT SIDES. ON EACH SIDE THE POINTS RUN FROM 1 TO
C   *      EITHER IMAX OR JMAX, AS APPROPRIATE.
C   *
C   *                *** FORMAT #2 (IFORM=2) ***
C   *
C   *      WRITE(10,1) C1
C   *      WRITE(10,1) C2
C   *      WRITE(10,1) IMAX,JMAX,NBSEG,NRSEG,LISEG,NBDY
C   *      WRITE(10,1) (LBSID(L),LB1(L),LB2(L),LBDY(L),LBEN(L),
C   *     1 L=1,NBSEG)
C   *      WRITE(10,1) (LRSID(L),LR1(L),LR2(L),LISID(L),LI1(L),LI2(L),
C   *     1 LTYPE(L),L=1,NRSEG)
C   *      DO 2 J=1,JMAX
C   *    2 WRITE(10,1) ((C(N,I,J),N=1,10),I=1,IMAX)
C   *
C   *      THE ARRAY C CONTAINS THE FACTORS FROM THE FIRST POINT TO THE
C   *      LAST POINT, ON THE FIRST ROW TO THE LAST ROW.
C   *
```

```
C   *                ***                           ***
C   *
C   *      THE FACTORS CORRESPOND TO THE INDEX N AS FOLLOWS:
C   *
C   *            N=1  : JACOBIAN
C   *            N=2  : ALPHA
C   *            N=3  : BETA
C   *            N=4  : GAMMA
C   *            N=5  : SIGMA
C   *            N=6  : TAU
C   *            N=7  : DX/DXI
C   *            N=8  : DX/DETA
C   *            N=9  : DY/DXI
C   *            N=10 : DY/DETA
C   *
C   ************************** INPUT DATA ***************************
C   *
C   *** CARD 1 IWRT1,IWRT2,IFORM = FORMAT(3I5)
C   *
C   *   IWRT1 = =0 DON'T PRINT COORDINATE SYSTEM FROM WHICH
C   *               FACTORS ARE CALCULATED.
C   *           >0 PRINT COORDINATE SYSTEM.
C   *
C   *   IWRT2 = =0 DON'T PRINT FACTORS.
C   *           >0 PRINT FACTORS.
C   *
C   *   IFORM = FILE STORAGE FORMAT CONTROL. (SEE ABOVE)
C   *
C   ****************************************************************
C
C   READ INPUT DATA
C
      WRITE(6,180)
      N1=1
      REWIND N1
      READ (5,110) IWRT1,IWRT2,IFORM
      WRITE (6,150) IWRT1,IWRT2,IFORM
      IF (IFORM.LT.1.OR.IFORM.GT.2) WRITE (6,170)
      IF (IFORM.LT.1.OR.IFORM.GT.2) STOP 1
C
C   READ COORDINATE SYSTEM
C
      READ (N1,101) C1
      READ (N1,101) C2
      READ (N1,110) IMAX,JMAX,NRSEG,NRBFG,LTSEG,NRDY
      ITEST=IMAX-2
      JTEST=JMAX-2
      IF (IFORM.EQ.2) ITEST=IMAX
      IF (IFORM.EQ.2) JTEST=JMAX
      IF (ITEST.GT.NDIM) WRITE (6,130)
      IF (JTEST.GT.NDIM1) WRITE (6,140)
      IF (ITEST.GT.NDIM.OR.JTEST.GT.NDIM1) STOP 2
      IF (MAX0(IMAX,JMAX).GT.NDIMX) WRITE (6,160)
      IF (MAX0(IMAX,JMAX).GT.NDIMX) STOP 3
      READ (N1,102)(LBSID(L),LB1(L),LB2(L),LBDY(L),LBEN(L),L=1,NBSEG)
      READ(N1,103)(LRSID(L),LR1(L),LR2(L),LTSID(L),LT1(L),LT2(L),
     1LTYPE(L),L=1,NRSEG)
      READ(N1,104)((X(I,J),I=1,IMAX),J=1,JMAX),((Y(I,J),I=1,IMAX),
     1J=1,JMAX)
C
```

```
C   PRINT LABEL
C
      WRITE (6,100) C1
      WRITE (6,100) C2
      WRITE (6,120) IMAX,JMAX
C
C   CALCULATE FACTORS
C
      JMXM1=JMAX-1
      IMXM1=IMAX-1
      MDIMX=MAX0(IMAX,JMAX)
      IMXM2=IMAX-2
      JMXM2=JMAX-2
      CALL ABG (C,X,Y,MDIM,IMAX,JMAX,IMXM1,JMXM1,IPRT2,NBSEG,LISEG,CH,LB
     1SID,LB1,LB2,LBDY,LPSID,LP1,LP2,LISID,LI1,LI2,LTYPE,LSEN,MDIMX,NBSE
     2G,MDIM1)
C
C   PRINT X AND Y FIELDS
C
      IF (IPRT1.EQ.0) GO TO 10
      CALL WRDATA (X,IMAX,JMAX,1,2,MDIM)
      CALL WRDATA (Y,IMAX,JMAX,3,4,MDIM)
C
C   WRITE FACTORS TO DISK
C
   10 GO TO (20,50), IFORM
C
C   *** FORMAT #1 ***
C
   20 WRITE(10,101) C1
      WRITE(10,101) C2
      WRITE(10,110) IMAX,JMAX,NBSEG,NPSEG,LISEG,NBDY
      WRITE(10,102)(LBSID(L),LB1(L),LB2(L),LBDY(L),LSEN(L),L=1,NBSEG)
      WRITE(10,103)(LPSID(L),LP1(L),LP2(L),LISID(L),LI1(L),LI2(L),
     1LTYPE(L),L=1,NPSEG)
      DO 30 J=1,JMXM2
   30 WRITE(10,104) ((C(N,I,J),N=1,10),I=1,IMXM2)
      DO 40 J=1,4
   40 WRITE(10,104)((CB(N,I,J),N=1,10),I=1,MDIMX)
      STOP 0101
C
C   *** FORMAT #2 ***
C
   50 WRITE(10,101) C1
      WRITE(10,101) C2
      WRITE(10,110) IMAX,JMAX,NBSEG,NPSEG,LISEG,NBDY
      WRITE(10,102)(LBSID(L),LB1(L),LB2(L),LBDY(L),LSEN(L),L=1,NBSEG)
      WRITE(10,103)(LPSID(L),LP1(L),LP2(L),LISID(L),LI1(L),LI2(L),
     1LTYPE(L),L=1,NPSEG)
      DO 60 JJ=1,JMXM2
        DO 60 II=1,IMXM2
          DO 60 N=1,10
            JJJ=JMXM1-JJ
            III=IMXM1-II
            J=JMAX-JJ
            I=IMAX-II
   60       C(N,I,J)=C(N,III,JJJ)
      DO 70 I=1,IMAX
        DO 70 N=1,10
          C(N,I,1)=CB(N,I,1)
```

```
   70       C(N,I,JMAX)=CB(N,I,3)
      DO 80 J=2,JMXM1
        DO 80 N=1,10
          C(N,1,J)=CB(N,J,2)
   80     C(N,IMAX,J)=CB(N,J,4)
      DO 90 J=1,JMAX
   90 WRITE(10,104) ((C(N,I,J),N=1,10),I=1,IMAX)
      STOP 0102
C
  100 FORMAT(1X,8A10)
  101 FORMAT(8A10)
  102 FORMAT(5I5)
  103 FORMAT(7I5)
  104 FORMAT(8E16.8)
  110 FORMAT (16I5)
  120 FORMAT (*0FIELD : IMAX =*,I4,5X,*JMAX =*,I4)
  130 FORMAT (*0****** ERROR ******,10X,*IMAX TOO LARGE, INCREASE*,* NDIM
     1 AND FIRST DIMENSION OF X,Y AND SECOND DIMENSION OF*,* C ,*)
  140 FORMAT (*0****** ERROR ******,10X,*JMAX TOO LARGE, INCREASE*,* NDIM
     11 AND SECOND DIMENSION OF X,Y AND THIRD*,* DIMENSION OF C, *)
  150 FORMAT (*0INPUT :  INRT1=*,I2,5X,*INRT2=*,I2,5X,*IFORM=*,I2)
  160 FORMAT (*0****** ERROR ******,10X,*MAXIMUM OF IMAX AND JMAX MUST*,*
     1 NOT BE GREATER THAN SECOND DIMENSION OF CB, INCREASE THIS*,* DIME
     2NSION*/
     3* AND NDIMX IN DATA STATEMENT,*)
  170 FORMAT (*0****** ERROR ******,10X,*IFORM MUST BE 1 OR 2,*)
  180 FORMAT(1H1//)
C
      END




      SUBROUTINE ABG (CF,X,Y,NDIM,IMAX,JMAX,IMXM1,JMXM1,INRT2,NBSEG,
     1LISEG,C,LBSID,LB1,LB2,LBDY,LRSID,LR1,LR2,LISID,LI1,LI2,LTYPE,
     2LSEN,NDIMX,NBSEG,NDIM1)
C
C  ************************* SCALE FACTORS ****************************
C  *
C  *******************************************************************
C
      DIMENSION X(NDIM,1), Y(NDIM,1), CF(10,NDIM,NDIM1), C(10,NDIMX,4)
      DIMENSION LBSID(1), LB1(1), LB2(1), LBDY(1), LRSID(1), LR1(1), LR2
     1(1), LISID(1), LI1(1), LI2(1), LTYPE(1), LSEN(1)
      DIMENSION FAC(10), SIDE(4)
      INTEGER FAC,SIDE
      REAL JCB
      DATA FAC /6HJACOBN,6HALPHA ,6HBETA  ,6HGAMMA ,6HSIGMA ,6HTAU
     16HX,XI  ,6HX,ETA ,6HY,XI  ,6HY,ETA /
      DATA SIDE/6HBOTTOM,6HLEFT  ,6HTOP   ,6HRIGHT /
C*****
      GO TO 260
C
C**** BODY SEGMENTS ****
C
   10 DO 100 L=1,NBSEG
        I1=LB1(L)
        I2=LB2(L)
        I3=I1+1
        I4=I2-1
```

```
      IGOTO=LHBID(L)
      GO TO (20,40,60,80), IGOTO
C***** BOTTOM
   20 CALL PARA3 (X,Y,I1,1,NDIM,XXI,YXI,I1,1,I1+1,1,I1+2,1,XETA,YETA,I
     11,1,I1,2,I1,3,1,1,BETA,GAMA,IGOTO,JCB,ALPHA)
      J=1
      I=I1
      JC=1
      IC=I1
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=BIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
      CALL PARA3 (X,Y,I2,1,NDIM,XXI,YXI,I2,1,I2-1,1,I2-2,1,XETA,YETA,I
     12,1,I2,2,I2,3,-1,1,BETA,GAMA,IGOTO,JCB,ALPHA)
      I=I2
      IC=I2
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=BIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
      DO 30 I=I3,I4
        CALL PARA3 (X,Y,I,1,NDIM,XXI,YXI,I-1,1,0,0,I+1,1,XETA,YETA,I,1
     1,I,2,I,3,0,1,BETA,GAMA,IGOTO,JCB,ALPHA)
        IC=I
        C(1,IC,JC)=JCB
        C(2,IC,JC)=ALPHA
        C(3,IC,JC)=BETA
        C(4,IC,JC)=GAMA
        C(5,IC,JC)=BIG
        C(6,IC,JC)=TAU
        C(7,IC,JC)=XXI
        C(8,IC,JC)=XETA
        C(9,IC,JC)=YXI
        C(10,IC,JC)=YETA
   30   CONTINUE
      GO TO 100
C***** LEFT
   40 CALL PARA3 (X,Y,1,I1,NDIM,XXI,YXI,1,I1,2,I1,3,I1,XETA,YETA,1,I1,
     11,I1+1,1,I1+2,1,1,BETA,GAMA,IGOTO,JCB,ALPHA)
      I=1
      J=I1
      JC=2
      IC=I1
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
```

```
      C(5,IC,JC)=BIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
      CALL PARAS (X,Y,1,I2,NDIM,XXI,YXI,1,I2,2,I2,3,I2,XETA,YETA,1,I2,
     11,I2-1,1,I2-2,1,-1,BETA,GAMA,IGOTO,JCB,ALPHA)
      J=I2
      IC=I2
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=BIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
      DO 50 I=I3,I4
        CALL PARAS (X,Y,1,I,NDIM,XXI,YXI,1,I,2,I,3,I,XETA,YETA,1,I-1,0
     1,0,1,I+1,1,0,BETA,GAMA,IGOTO,JCB.ALPHA)
        IC=I
        C(1,IC,JC)=JCB
        C(2,IC,JC)=ALPHA
        C(3,IC,JC)=BETA
        C(4,IC,JC)=GAMA
        C(5,IC,JC)=BIG
        C(6,IC,JC)=TAU
        C(7,IC,JC)=XXI
        C(8,IC,JC)=XETA
        C(9,IC,JC)=YXI
        C(10,IC,JC)=YETA
50      CONTINUE
      GO TO 100
C**** TOP
60    CALL PARAS (X,Y,I1,JMAX,NDIM,XXI,YXI.I1,JMAX,I1+1.JMAX,I1+2,JMAX
     1,XETA,YETA,I1.JMAX,I1.JMAX-1,I1,JMAX-2,1,-1,BETA,GAMA,IGOTO,JCB,AL
     2PHA)
      J=JMAX
      I=I1
      JC=3
      IC=I1
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=BIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
      CALL PARAS (X,Y,I2,JMAX,NDIM,XXI,YXI.I2,JMAX,I2-1.JMAX,I2-2,JMAX
     1,XETA,YETA.I2,JMAX,I2.JMAX-1,I2,JMAX-2,-1,-1.BETA,GAMA,IGOTO,JCB,A
     2LPHA)
      I=I2
      IC=I2
      C(1,IC,JC)=JCB
```

```
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=SIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
      DO 70 I=I3,I4
        CALL PARA3 (X,Y,I,JMAX,NDIM,XXI,YXI,I-1,JMAX,0,0,I+1,JMAX,XETA
     1,YETA,I,JMAX,I,JMAX-1,I,JMAX-2,0,-1,BETA,GAMA,IGOTO,JCB,ALPHA)
        IC=I
        C(1,IC,JC)=JCB
        C(2,IC,JC)=ALPHA
        C(3,IC,JC)=BETA
        C(4,IC,JC)=GAMA
        C(5,IC,JC)=SIG
        C(6,IC,JC)=TAU
        C(7,IC,JC)=XXI
        C(8,IC,JC)=XETA
        C(9,IC,JC)=YXI
        C(10,IC,JC)=YETA
   70   CONTINUE
      GO TO 100
C**** RIGHT
   80 CALL PARA3 (X,Y,IMAX,I1,NDIM,XXI,YXI,IMAX,I1,IMAX-1,I1,IMAX-2,I1
     1,XETA,YETA,IMAX,I1,IMAX,I1+1,IMAX,I1+2,-1,1,BETA,GAMA,IGOTO,JCB,AL
     2PHA)
      I=IMAX
      J=I1
      JC=4
      IC=I1
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=SIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
      CALL PARA3 (X,Y,IMAX,I2,NDIM,XXI,YXI,IMAX,I2,IMAX-1,I2,IMAX-2,I2
     1,XETA,YETA,IMAX,I2,IMAX,I2-1,IMAX,I2-2,-1,-1,BETA,GAMA,IGOTO,JCB,A
     2LPHA)
      J=I2
      IC=I2
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=SIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
      DO 90 I=I3,I4
        CALL PARA3 (X,Y,IMAX,I,NDIM,XXI,YXI,IMAX,I,IMAX-1,I,IMAX-2,I,X
```

```
     1ETA,YETA,IMAX,I+1,0,0,IMAX,I+1,-1,0,BETA,GAMA,IGOTO,JCB,ALPHA)
            IC=I
            C(1,IC,JC)=JCB
            C(2,IC,JC)=ALPHA
            C(3,IC,JC)=BETA
            C(4,IC,JC)=GAMA
            C(5,IC,JC)=BIG
            C(6,IC,JC)=TAU
            C(7,IC,JC)=XXI
            C(8,IC,JC)=XETA
            C(9,IC,JC)=YXI
            C(10,IC,JC)=YETA
   90       CONTINUE
  100    CONTINUE
C
C**** REENTRANT SEGMENTS ****
C
      IF (NRSEG.EQ.0) GO TO 245
      DO 240 L=1,NRSEG
         I5=LR1(L)+1
         I6=LR2(L)-1
         I3=LI1(L)+1
         I4=LI2(L)-1
         IGOTO=LTYPE(L)
         GO TO (110,130,150,170,190,210), IGOTO
C**** ONE ON BOTTOM, ONE ON TOP
  110    DO 120 I=I5,I6
            J=1
            XXI=(X(I+1,1)-X(I-1,1))*0.5
            YXI=(Y(I+1,1)-Y(I-1,1))*0.5
            XETA=(X(I,2)-X(I,JMXM1))*0.5
            YETA=(Y(I,2)-Y(I,JMXM1))*0.5
            I1=I-1
            I2=I+1
            J1=JMXM1
            J2=2
            CALL PARA1 (XXIS,YXIS,I2,J,I1,J,XETAS,YETAS,I,J2,I,J1,XXIETA,Y
     1XIETA,I2,J2,I2,J1,I1,J1,I1,J2,X,Y,I,J,NDIM)
            CALL PARA2 (XXI,YXI,XETA,YETA,XXIS,YXIS,XETAS,YETAS,XXIETA,YXI
     1ETA,ALPHA,GAMA,BETA,BIG,TAU,JCB,I,J,NDIM)
            JC=1
            IC=I
            C(1,IC,JC)=JCB
            C(2,IC,JC)=ALPHA
            C(3,IC,JC)=BETA
            C(4,IC,JC)=GAMA
            C(5,IC,JC)=BIG
            C(6,IC,JC)=TAU
            C(7,IC,JC)=XXI
            C(8,IC,JC)=XETA
            C(9,IC,JC)=YXI
            C(10,IC,JC)=YETA
  120       CONTINUE
         GO TO 230
C**** BOTH ON BOTTOM
  130    DO 140 I=I5,I6
            J=1
            II=I4-(I-I5)
            XXI=(X(I+1,1)-X(I-1,1))*0.5
            YXI=(Y(I+1,1)-Y(I-1,1))*0.5
```

```
      XETA=(X(I,2)-X(II,2))*0.5
      YETA=(Y(I,2)-Y(II,2))*0.5
      I1=I-1
      I2=I+1
      J1=II
      J2=2
      CALL PARA1 (XXIS,YXIS,I2,J,I1,J,XETAS,YETAS,I,J2,J1,J2,XXIETA,
     1YXIETA,I2,J2,J1-1,J2,J1+1,J2,I1,J2,X,Y,T,J,NDIM)
      CALL PARA2 (XXI,YXI,XETA,YETA,XXIS,YXIS,XETAS,YETAS,XXIETA,YXI
     1ETA,ALPHA,GAMA,BETA,SIG,TAU,JCB,I,J,NDIM)
      JC=1
      IC=I
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=SIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
140   CONTINUE
    GO TO 230
C**** BOTH ON TOP
150 DO 160 I=I5,I6
      J=JMAX
      II=I4-(I-I5)
      XXI=(X(I+1,JMAX)-X(I-1,JMAX))*0.5
      YXI=(Y(I+1,JMAX)-Y(I-1,JMAX))*0.5
      XETA=(X(II,JMXM1)-X(I,JMXM1))*0.5
      YETA=(Y(II,JMXM1)-Y(I,JMXM1))*0.5
      I1=I-1
      I2=I+1
      J1=II
      J2=JMXM1
      CALL PARA1 (XXIS,YXIS,I2,J,I1,J,XETAS,YETAS,J1,J2,I,J2,XXIETA,
     1YXIETA,J1-1,J2,I2,J2,I1,J2,J1+1,J2,X,Y,T,J,NDIM)
      CALL PARA2 (XXI,YXI,XETA,YETA,XXIS,YXIS,XETAS,YETAS,XXIETA,YXI
     1ETA,ALPHA,GAMA,BETA,SIG,TAU,JCB,I,J,NDIM)
      JC=3
      IC=I
      C(1,IC,JC)=JCB
      C(2,IC,JC)=ALPHA
      C(3,IC,JC)=BETA
      C(4,IC,JC)=GAMA
      C(5,IC,JC)=SIG
      C(6,IC,JC)=TAU
      C(7,IC,JC)=XXI
      C(8,IC,JC)=XETA
      C(9,IC,JC)=YXI
      C(10,IC,JC)=YETA
160   CONTINUE
    GO TO 230
C**** ONE ON LEFT, ONE ON RIGHT
170 DO 180 J=I5,I6
      I=1
      XXI=(X(2,J)-X(IMXM1,J))*0.5
      YXI=(Y(2,J)-Y(IMXM1,J))*0.5
      XETA=(X(1,J+1)-X(1,J-1))*0.5
```

```
          YETA=(Y(1,J+1)-Y(1,J-1))*0.5
          I1=IMXM1
          I2=2
          J1=J-1
          J2=J+1
          CALL PARA1 (XXIS,YXIS,I2,J,I1,J,XETAS,YETAS,I,J2,I,J1,XXIETA,Y
     1XIETA,I2,J2,I2,J1,I1,J1,I1,J2,X,Y,I,J,NDIM)
          CALL PARA2 (XXI,YXI,XETA,YETA,XXIS,YXIS,XETAS,YETAS,XXIETA,YXI
     1ETA,ALPHA,GAMA,BETA,SIG,TAU,JCB,I,J,NDIM)
          JC=2
          IC=J
          C(1,IC,JC)=JCB
          C(2,IC,JC)=ALPHA
          C(3,IC,JC)=BETA
          C(4,IC,JC)=GAMA
          C(5,IC,JC)=SIG
          C(6,IC,JC)=TAU
          C(7,IC,JC)=XXI
          C(8,IC,JC)=XETA
          C(9,IC,JC)=YXI
          C(10,IC,JC)=YETA
  180     CONTINUE
        GO TO 230
C**** BOTH ON LEFT
  190   DO 200 J=I5,I6
          I=1
          II=I4-(J-I5)
          XXI=(X(2,J)-X(2,II))*0.5
          YXI=(Y(2,J)-Y(2,II))*0.5
          XETA=(X(1,J+1)-X(1,J-1))*0.5
          YETA=(Y(1,J+1)-Y(1,J-1))*0.5
          I1=II
          I2=2
          J1=J-1
          J2=J+1
          CALL PARA1 (XXIS,YXIS,I2,J,I2,I1,XETAS,YETAS,I,J2,I,J1,XXIETA,
     1YXIETA,I2,J2,I2,J1,I2,I1+1,I2,I1-1,X,Y,I,J,NDIM)
          CALL PARA2 (XXI,YXI,XETA,YETA,XXIS,YXIS,XETAS,YETAS,XXIETA,YXI
     1ETA,ALPHA,GAMA,BETA,SIG,TAU,JCB,I,J,NDIM)
          JC=2
          IC=J
          C(1,IC,JC)=JCB
          C(2,IC,JC)=ALPHA
          C(3,IC,JC)=BETA
          C(4,IC,JC)=GAMA
          C(5,IC,JC)=SIG
          C(6,IC,JC)=TAU
          C(7,IC,JC)=XXI
          C(8,IC,JC)=XETA
          C(9,IC,JC)=YXI
          C(10,IC,JC)=YETA
  200     CONTINUE
        GO TO 230
C**** BOTH ON RIGHT
  210   DO 220 J=I5,I6
          I=IMAX
          II=I4-(J-I5)
          XXI=(X(IMXM1,II)-X(IMXM1,J))*0.5
          YXI=(Y(IMXM1,II)-Y(IMXM1,J))*0.5
          XETA=(X(IMAX,J+1)-X(IMAX,J-1))*0.5
```

```
          YETA=(Y(IMAX,J+1)-Y(IMAX,J-1))*0.5
          I1=II
          I2=IMXM1
          J1=J-1
          J2=J+1
          CALL PARA1 (XXIS,YXIS,I2,I1,I2,J,XETAS,YETAS,I,J2,I,J1,XXIETA,
     1YXIETA,I2,I1-1,I2,I1+1,I2,J1,I2,J2,X,Y,I,J,NDIM)
          CALL PARA2 (XXI,YXI,XETA,YETA,XXIS,YXIS,XETAS,YETAS,XXIETA,YXI
     1ETA,ALPHA,GAMA,BETA,SIG,TAU,JCB,I,J,NDIM)
          JC=4
          IC=J
          C(1,IC,JC)=JCB
          C(2,IC,JC)=ALPHA
          C(3,IC,JC)=BETA
          C(4,IC,JC)=GAMA
          C(5,IC,JC)=SIG
          C(6,IC,JC)=TAU
          C(7,IC,JC)=XXI
          C(8,IC,JC)=XETA
          C(9,IC,JC)=YXI
          C(10,IC,JC)=YETA
  220     CONTINUE
  230   CONTINUE
  240   CONTINUE
  245 DO 246 N=1,10
      TEM = C(N,1,1) + C(N,1,2)
      C(N,1,1) = TEM
      C(N,1,2) = TEM
      TEM = C(N,JMAX,2) + C(N,1,3)
      C(N,JMAX,2) = TEM
      C(N,1,3) = TEM
      TEM = C(N,IMAX,3) + C(N,JMAX,4)
      C(N,IMAX,3) = TEM
      C(N,JMAX,4) = TEM
      TEM = C(N,1,4) + C(N,IMAX,1)
      C(N,1,4) = TEM
      C(N,IMAX,1) = TEM
  246 CONTINUE
      IF (IWRT2.EQ.0) GO TO 290
      WRITE (6,330)
      DO 250 J=1,4
        WRITE (6,340) SIDE(J)
      IF (J.EQ.1) IMAXX1 = IMAX
      IF (J.EQ.3) IMAXX1 = IMAX
      IF(J.EQ.2) IMAXX1 = JMAX
      IF(J.EQ.4) IMAXX1 = JMAX
        DO 250 N=1,10
  250   WRITE(6,320) FAC(N),(C(N,I,J),I=1,IMAXX1)
      GO TO 290
C
C**** FIELD ****
C
  260 CONTINUE
      DO 270 J=2,JMXM1
        JC=J-1
        JM1=J-1
        JP1=J+1
        DO 270 I=2,IMXM1
          IC=I-1
          IP1=I+1
```

```
            IM1=I-1
            XETA=(X(I,JP1)-X(I,JM1))*0.5
            YETA=(Y(I,JP1)-Y(I,JM1))*0.5
            XXI=(X(IP1,J)-X(IM1,J))*0.5
            YXI=(Y(IP1,J)-Y(IM1,J))*0.5
            ALPHA=XETA**2+YETA**2
            BETA=XXI*XETA+YXI*YETA
            GAMA=XXI**2+YXI**2
            JCB=XXI*YETA-XETA*YXI
            XXIS=X(IP1,J)-2.0*X(I,J)+X(IM1,J)
            YXIS=Y(IP1,J)-2.0*Y(I,J)+Y(IM1,J)
            XETAS=X(I,JP1)-2.0*X(I,J)+X(I,JM1)
            YETAS=Y(I,JP1)-2.0*Y(I,J)+Y(I,JM1)
            XXIETA=0.25*(X(IP1,JP1)-X(IM1,JP1)-X(IP1,JM1)+X(IM1,JM1))
            YXIETA=0.25*(Y(IP1,JP1)-Y(IM1,JP1)-Y(IP1,JM1)+Y(IM1,JM1))
            DX=ALPHA*XXIS-2.0*BETA*XXIETA+GAMA*XETAS
            DY=ALPHA*YXIS-2.0*BETA*YXIETA+GAMA*YETAS
            SIG=(DX*YXI-DY*XXI)/JCB
            TAU=(DY*XETA-DX*YETA)/JCB
            ABSJ=ABS(JCB)
            CF(1,IC,JC)=JCB
            CF(2,IC,JC)=ALPHA
            CF(3,IC,JC)=BETA
            CF(4,IC,JC)=GAMA
            CF(5,IC,JC)=SIG
            CF(6,IC,JC)=TAU
            CF(7,IC,JC)=XXI
            CF(8,IC,JC)=XETA
            CF(9,IC,JC)=YXI
            CF(10,IC,JC)=YETA
  270       CONTINUE
C
C  WRITE FACTORS TO PRINTER
C
      IF (IWRT2.EQ.0) GO TO 10
      WRITE (6,300)
      JMXM2=JMAX-2
      IMXM2=IMAX-2
      DO 280 J=1,JMXM2
      JJ=J+1
        WRITE (6,310) JJ
        DO 280 N=1,10
  280     WRITE (6,320) FAC(N),(CF(N,I,J),I=1,IMXM2)
      GO TO 10
  290 CONTINUE
      RETURN
C
  300 FORMAT ('0***** SCALE FACTORS *****')
  310 FORMAT (//  ' J =',I3/)
  320 FORMAT ('0',A6//(8E15.8))
  330 FORMAT (//  ' BOUNDARY')
  340 FORMAT (//2H ',A6)
C
      END




      FUNCTION AMXMN (NOPT,A,AX,I,J,IX,JX)
```

```
C
C     ******************************************************************
C     *
C     *   NOPT=1 : A AND AX ARE COMPARED FOR MAXIMUM VALUE.
C     *   NOPT=2 : A AND AX ARE COMPARED FOR MAXIMUM VALUE.
C     *
C     ******************************************************************
C
      GO TO (10,20), NOPT
   10 IF (A.LT.AX) GO TO 40
      GO TO 30
   20 IF (A.GT.AX) GO TO 40
   30 AMXMN=A
      IX=I
      JX=J
      RETURN
   40 AMXMN=AX
      RETURN
C
      END




      SUBROUTINE PARA1 (X1,Y1,I1,J1,I2,J2,X2,Y2,I3,J3,I4,J4,X12,Y12,I5,J
     15,I6,J6,I7,J7,I8,J8,X,Y,I,J,NDIM)
C
C     ****************** SECOND DERIVATIVES ******************************
C     *
C     ******************************************************************
C
      DIMENSION X(NDIM,1), Y(NDIM,1)
C*****
      X1=X(I1,J1)-2.0*X(I,J)+X(I2,J2)
      Y1=Y(I1,J1)-2.0*Y(I,J)+Y(I2,J2)
      X2=X(I3,J3)-2.0*X(I,J)+X(I4,J4)
      Y2=Y(I3,J3)-2.0*Y(I,J)+Y(I4,J4)
      X12=0.25*(X(I5,J5)-X(I6,J6)+X(I7,J7)-X(I8,J8))
      Y12=0.25*(Y(I5,J5)-Y(I6,J6)+Y(I7,J7)-Y(I8,J8))
      RETURN
C
      END




      SUBROUTINE PARA2 (XXI,YXI,XETA,YETA,XXIS,YXIS,XETAS,YETAS,XXIETA,Y
     1XIETA,ALPHA,GAMA,BETA,SIG,TAU,JCB,I,J,NDIM)
C
C     ******************** FACTORS ON RE-ENTRANT SEGMENTS ***************
C     *
C     ******************************************************************
C
      REAL JCB
C*****
      ALPHA=XETA**2+YETA**2
      BETA=XXI*XETA+YXI*YETA
      GAMA=XXI**2+YXI**2
      JCB=XXI*YETA-XETA*YXI
      DX=ALPHA*XXIS-2.0*BETA*XXIETA+GAMA*XETAS
```

```
      DY=ALPHA*YXIS-2.0*BETA*YXIETA+GAMA*YETAS
      SIG=(DX*YXI-DY*XXI)/JCB
      TAU=(DY*XETA-DX*YETA)/JCB
      RETURN
C
      END




      SUBROUTINE PARAS (X,Y,I,J,NDIM,XXI,YXI,I1,J1,I2,J2,I3,J3,XETA,YETA
     1,I4,J4,I5,J5,I6,J6,K1,K2,BETA,GAMA,IGOTO,JCB,ALPHA)
C
C   ************* ALPHA, BETA, GAMMA, JACOBIAN *************************
C   *
C   *******************************************************************
C
      DIMENSION X(NDIM,1), Y(NDIM,1)
      REAL JCB
C*****
      IF (K1.EQ.0) GO TO 10
      XXI=-0.5*(X(I3,J3)-4.0*X(I2,J2)+3.0*X(I1,J1))*K1
      YXI=-0.5*(Y(I3,J3)-4.0*Y(I2,J2)+3.0*Y(I1,J1))*K1
      GO TO 20
   10 XXI=0.5*(X(I3,J3)-X(I1,J1))
      YXI=0.5*(Y(I3,J3)-Y(I1,J1))
   20 IF (K2.EQ.0) GO TO 30
      XETA=-0.5*(X(I6,J6)-4.0*X(I5,J5)+3.0*X(I4,J4))*K2
      YETA=-0.5*(Y(I6,J6)-4.0*Y(I5,J5)+3.0*Y(I4,J4))*K2
      GO TO 40
   30 XETA=0.5*(X(I6,J6)-X(I4,J4))
      YETA=0.5*(Y(I6,J6)-Y(I4,J4))
   40 CONTINUE
      ALPHA=XETA**2+YETA**2
      BETA=XXI*XETA+YXI*YETA
      GAMA=XXI**2+YXI**2
      JCB=XXI*YETA-XETA*YXI
      RETURN
C
      END




      SUBROUTINE WRDATA (X,IMAX,JMAX,I1,I2,NDIM)
C
C   ************************* PRINT SOLUTION ***************************
C   *
C   *******************************************************************
C
      DIMENSION CODE(4)
      DIMENSION X(NDIM,1)
      DATA CODE/6H** XAR,6HRAY **,6H** YAR,6HRAY **/
C*****
      WRITE (6,20) (CODE(I),I=I1,I2)
      DO 10 J=1,JMAX
        WRITE (6,30) J
   10   WRITE (6,40) (X(I,J),I=1,IMAX)
      RETURN
C
```

```
   20 FORMAT (1H1,20X,2A6//)
   30 FORMAT (5X,*J=*I4)
   40 FORMAT (4X,10E11.5)
C
      END
```

SAMPLE CASE INPUT: SINGLE-BODY FIELD

Input for Program FATCAT requires only one card and is described beginning at line number 76 in FATCAT listing. For the sample case computed:

```
      1     1     2
```

Disk file 1 needed as input is TAPE 10 saved from the TOMCAT program.

Sample Case Output: Single-Body Field

```
INPUT 1  [illegible] 1     [illegible] 1     [illegible] 2
TEST CASE - BODY-FITTED COORDINATE SYSTEM
SINGLE BODY : KARMAN-TREFFTZ AIRFOIL, 26 POINTS

FIELD : IMAX = 27     JMAX = 26

**** SCALE FACTORS ****

J = 2

JACOBN

[illegible]

ALPHA

[illegible]

BETA

[illegible]

GAMMA

[illegible]

SIGMA

[illegible]

TAU

[illegible]

X,XI

[illegible]

Y,ETA

[illegible]
```

```
Y,XI

-.79309124E-02 -.94250124E-02 -.11779035E-01 -.15549988E-01 -.17420842E-01 -.14908308E-01 -.10465013E-01 -.30282720E-02
 .60803940E-02  .16438348E-01  .26320160E-01  .33544408E-01  .36955347E-01  .35077260E-01  .29022077E-01  .19855260E-01
 .79627200E-02 -.38085750E-02 -.13991752E-01 -.20983958E-01 -.23753128E-01 -.19914012E-01 -.13099701E-01 -.10418129E-01
-.83173241E-02

Y,ETA

-.78915415E-02 -.15879055E-01 -.24219112E-01 -.33656052E-01 -.40440800E-01 -.44874645E-01 -.44579845E-01 -.45895450E-01
-.42839475E-01 -.38838375E-01 -.27111253E-01 -.13570549E-01  .23121170E-02  .17411315E-01  .29272435E-01  .37352031E-01
 .42245572E-01  .44805454E-01  .45520150E-01  .44317775E-01  .40714260E-01  .34614001E-01  .24230099E-01  .15421907E-01
 .71100139E-02
```

⋮

[illegible]

[illegible]

[illegible]

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

ALPHA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

DELTA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

GAMMA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

SIGMA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

TAU

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

X,XI

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

X,ETA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

Y,XI

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

TOP

JACOB*

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

ALPHA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

BETA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

GAMMA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

SIGMA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

TAU

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

X,XI

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0.	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

X,ETA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

Y,XI

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

Y,ETA

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]					

```
WEIGHT

JACOBN

 .4994481942E+05 0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.              .1040418948E+01

ALPHA

 .1311398271E+04 0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.              .7416470 8E+00

BETA

 .1139462596E+06 0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             -.5168895405E-02

GAMMA

 .1903454456E+05 0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.              .1516237388E+01

SIGMA

-.3373082081E+04 0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             -.3373082081E+04

TAU

 .3203126838E+04 0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.              .3203126838E+04

X,XI

-.5859000000E+04 0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.              .4221850046E-02

X,ETA

 .3613915 [illegible]E-02 0.    0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.              .8011442 [illegible]E+00

Y,XI

-.137895 [illegible]E-02 0.     0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.             -.1431130871E+01

Y,ETA

-.2307 [illegible]E+15 0.       0.             0.             0.             0.             0.             0.
0.               0.             0.             0.             0.             0.             0.             0.
0.               0.             0.              .7 [illegible]5637 [illegible]E-02
```

REFERENCES

1. Winslow, A. J., "Numerical Solution of the Quasi-Linear Poisson Equation in a Non-Uniform Triangular Mesh," Journal of Computational Physics, 2, 149 (1966).
2. Barfield, W. D., "An Optimal Mesh Generator for Lagrangian Hydrodynamic Calculations in Two Space Dimensions," Journal of Computational Physics, 6, 417 (1970).
3. Chu, W. H., "Development of a General Finite Difference Approximation for a General Domain, Part I: Machine Transformation," Journal of Computational Physics, 8, 392 (1971).
4. Amsden, A. A. and Hirt, C. W., "A Simple Scheme for Generating General Curvilinear Grids," Journal of Computational Physics, 11, 348 (1973).
5. Godunov, S. K. and Prokopov, G. P., "The Use of Moving Meshes in Gas Dynamics Computations," USSR Computational Mathematics and Mathematical Physics, 12, 182 (1972).
6. Stadius, G., "Construction of Orthogonal Curvilinear Meshes by Solving Initial Value Problems," Dept. of Computer Sciences Report No. 53, Uppsala University, Sweden (1974).
7. Meyder, R., "Solving the Conservation Equations in Fuel Rod Bundles Exposed to Parallel Flow by Means of Curvilinear-Orthogonal Coordinates," Journal of Computational Physics, 17, 53 (1975).
8. Ives, D. C., "A Modern Look at Conformal Mapping, Including Doubly Connected Regions," AIAA Paper No. 75-842, AIAA 8th Fluid and Plasma Dynamics Conference, Hartford, (1975).
9. Gal-Chen, T. and Somerville, R. C. J., "On the Use of a Coordinate Transformation for the Solution of the Navier-Stokes Equations," Journal of Computational Physics, 17, 209 (1975).

10. Thompson, J. F., Thames, F. C., and Mastin, C. W., "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Field Containing Any Number of Arbitrary Two-Dimensional Bodies," Journal of Computational Physics, 15, 299, (1974).

11. Thames, Frank C., "Numerical Solution of the Incompressible Navier-Stokes Equations About Arbitrary Two-Dimensional Bodies," Ph.D. Dissertation, Mississippi State University, (1975).

12. Thames, F. C., Thompson, J. F., and Mastin, C. W., "Numerical Solution of the Navier-Stokes Equations for Arbitrary Two-Dimensional Airfoils," Proceedings of NASA Conference on Aerodynamic Analyses Requiring Advanced Computers, Langley Research Center, NASA SP-347, (1975).

13. Thompson, J. F., Thames, F. C., Mastin, C. W., and Shanks, S. P., "Numerical Solutions of the Unsteady Navier-Stokes Equations for Arbitrary Bodies Using Boundary-Fitted Curvilinear Coordinates," Proceedings of Arizona/AFOSR Symposium on Unsteady Aerodynamics, Univ. of Arizona, Tucson, Arizona, (1975).

14. Thompson, J. F., Thames, F. C., and Shanks, S. P., "Use of Numerically Generated Body-Fitted Coordinate Systems for Solution of the Navier-Stokes Equations," Proceedings of AIAA 2nd Computational Fluid Dynamics Conference, Hartford, Connecticut, (1975).

15. Kolman, B., and Trench, W. F., Elementary Multivariable Calculus, Academic Press, New York, (1970).

16. Levinson, N., and Redheffer, R. M., Complex Variables, Holden Day San Francisco, (1970).

17. Mastin, C. W., and Thompson, J. F., "Elliptic Systems and Numerical Transformations," ICASEReport 76-14, NASA Langley Research Center (1976).

18. Varga, R. L., Matrix Iterative Analysis, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, (1962).

19. Thompson, J. F., and Thames, F. C., "Numerical Solution for Potential Flow about Arbitrary Two-Dimensional Bodies Using Body-Fitted Coordinate Systems," NASA CR to be released 1976.

20. Karamacheti, K., Principles of Ideal-Fluid Aerodynamics, John Wiley & Sons, Inc., (1966).

21. Liebeck, R. H., Wind Tunnel Tests of Two Airfoils Designed for High Lift Without Separation in Incompressible Flow, McDonnel Douglas Corporation, Report No. MDC-J5667/01, (1972). See also, Liebeck, R. H., "A Class of Airfoils Designed for High Lift in Incompressible Flow," AIAA Paper 73-86, AIAA 11th Aerospace Sciences Meeting, Washington, 1973.

22. Morse, P. M. K., and Feshbach, H., Methods of Theoretical Physics, Vol. II, McGraw-Hill, (1953).

23. Hodge, J. K., "Numerical Solution in Natural Curvilinear Coordinates of Incompressible Laminar Flow Through Arbitrary Two-Dimensional and Axisymmetric Bodies," Ph.D. Dissertation, Mississippi State University, (1975).

24. McWhorter, J. C., Unpublished Research, Department of Aerophsycis and Aerospace Engineering, Mississippi State University, (1975).

25. Timoshenko, S., Theory of Plates and Shells, McGraw-Hill Book Company, Inc., 1940.

26. Thompson, J. F., Thames, F. C., Shanks, S. P., Reddy, R. N., and Mastin, C. W., "Solutions of the Navier-Stokes Equations in Various Flow Regimes on Fields Containing Any Number of Arbitrary Bodies Using Boundary-Fitted Coordinate Systems," Proceeding of V International Conference on Numerical Methods in Fluid Dynamics, to be published in Lecture Notes in Physics, Springer Verlag, (1976).

TABLE I

Input Parameters for Various Segment Arrangements

Figure	IMAX	JMAX	NBSEG	NRSEG	LBSID	LB1	LB2	LBDY	LRSID	LR1	LR2	LISID	LI1	LI2	NINF	Body #1	Body #2
4a, 14	37	20	2	1	1 3	1 1	37 37	1 0	2	1	20	4	1	20	36		
5, 15	81	21	4	2	1 1 1 3	26 66 1 1	56 81 15 81	1 2 2 0	1 2	16 1	26 21	1 4	56 1	66 21	80	30/270-U	30/90-L
16	71	16	4	2	1 3 2 4	21 51 1 16	51 21 16 1	1 2 0 0	1 1	1 51	21 71	3 3	1 51	21 71	30	30/270-L	30/90-U
17	56	31	4	2	4 2 1 3	1 31 36 21	31 1 21 36	1 2 0 0	1 1	1 36	21 56	3 3	1 36	21 56	30	30/270-L	30/90-U
8, 18	51	36	4	2	1 4 2 3	11 1 16 1	41 16 1 51	1 2 2 0	1 2	1 16	11 36	1 4	41 16	51 36	50	30/270-U	30/90-L
9, 19	31	46	4	2	1 4 2 3	1 11 26 1	31 26 11 31	1 2 2 0	2 2	1 26	11 46	4 4	1 26	11 46	30	30/270-U	30/90-L
10, 20	71	26	4	2	1 4 2 3	1 11 26 21	71 26 11 51	1 2 2 0	2 3	1 1	11 21	4 3	1 51	11 71	30	70/270-U	30/90-L

TABLE 1 Continued

Figure	IMAX	JMAX	NBSEG	NRSEG	LBSID	LB1	LB2	LBDY	LRSID	LR1	LR2	LISID	LI1	LI2	NINF	Body #1	Body #2
11, 21	101	11	4	2	1 3 3 3	1 101 16 36	101 86 1 66	1 2 2 0	2 3	1 16	11 36	4 3	1 66	11 86	30	100/270-U	30/90-L
12, 22	71	36	4	2	1 3 2 4	1 51 21 36	71 21 36 21	1 2 0 0	2 3	1 1	21 21	4 3	1 51	21 71	30	70/270-L	30/90-U
13, 23	101	21	4	2	1 3 3 3	1 66 1 86	101 36 16 101	1 2 0 0	2 3	1 16	21 36	4 3	1 66	21 86	30	100/270-L	30/90-U

NBDY	2
ITER	100
IGES	varied
R(1)	1.8
R(2)	0.0001
R(3)	0.0001
YINFIN	5.0
AINFIN	-90.0
XOINF	0.0
YOINF	0.0
IEV	0
R(10)	0.0
INFAC	0
INFACO	0

No attraction

NOTE: The first number in the body name is the number of points on the body. The second number is the angle (counter-clockwise from the x-axis) of the first point. The letter indicates a location above (U) or below (L) the x-axis.

TABLE 2

Effect of Initial Guess on Convergence

Configuration	IGES: 0	1	2	3	-1	-4	-10	-20	-40	-100
(Fig. 4a)	68	NC	80	NC	NC	NC	NC	--	69	71
(Fig. 5)	75	NC	64	NC	100+	100+	70	63	67	68
(Fig. 6)	NC	NC[1]	NC	NC	NC	NC	NC	NC	NC	NC
(Fig. 7)	NC	NC	NC	NC[2]	NC	NC	NC	NC	NC	NC
(Fig. 8)	100+	100+	100+	NC	NC	100+	100+	87	84	86
(Fig. 9)	100+	100+	100+	NC	NC	100+	NC	NC	NC	NC
(Fig. 10)	99	89	99	NC	NC	100+	91	84	82	86
(Fig. 11)	NC	100+	100+	NC	NC	100+	100	93	91	94
(Fig. 12)	NC	NC	NC	NC	NC	100+	100+	100+	100+	100+
(Fig. 13)	NC	NC	100+	NC	NC	100+	100+	100+	100+	99

Legend: Numbers given are the number of iterations required for convergence to 0.0001.

NC indicates no tendency toward convergence in 100 iterations.

100+ indicates a converging solution at 100 iterations.

Notes: [1]IGES = 1 gave convergence for the field shown in Figure 6.

[2]IGES = 4 (Guess 3 without division by total number of boundary points) gave convergence for the field shown in Figure 7.

TABLE 3

Input Parameters for Basic Single-Body Field Used in Optimum Acceleration Parameter Studies

Case 1

IMAX	67	
JMAX	40	
NBDY	1	
IGES	0	
NBSEG	2	
NRSEG	1	
LBSID	1	3
LB1	1	1
LB2	67	67
LBDY	1	0
LRSID	2	
LR1	1	
LR2	40	
LISID	4	
LI1	1	
LI2	40	
R(1)	varied	
R(2)	0.0001	
R(3)	0.0001	
YINFIN	10.0	
AINFIN	0.0	
XOINF	0.0	
YOINF	0.0	
NINF	66	

TABLE 3 Continued

IEV	0
R(10)	0.0
INFAC	0
INFACO	0
ATYP	ETA
ITYP	0
NLN	1
NPT	0
DEC	0.0
AMPFAC	0.0
JLN	1
ALN	1000.0
DLN	1.0
ATYP	0
ITYP	0
NLN	0
NPT	0
DEC	0.0
AMPFAC	0.0
IFAC	0
EFAC	100.0

TABLE 4

Input Parameters Varied for Single-Body Field Used in Optimum Acceleration Parameter Studies

Case	IMAX	JMAX	LB2	LR2	LI2	YINFIN	NINF	ALN	IFAC
2								100.0	
3								10000.0	4
4	37		37 37				36		
5	97		97 97				96		
6		20		20	20				
7		60		60	60				
8						5.0			
9						20.0			
10	37		37 37				36	100.0	
11	37		37 37				36	10000.0	4
12	97		97 97				96	100.0	
13	97		97 97				96	10000.0	4
14		20		20	20			100.0	
15		20		20	20			10000.0	4
16		60		60	60			100.0	
17		60		60	60			10000.0	4
18						5.0		100.0	
19						5.0		10000.0	4
20						20.0		100.0	
21						20.0		10000.0	4

TABLE 4 Continued

Case	IMAX	JMAX	LB2	LR2	LI2	YINFIN	NINF	ALN	IFAC
22	37	20	37 37	20	20		36		
23	97	20	97 97	20	20		96		
24	37	60	37 37	60	60		36		
25	97	60	97 97	60	60		96		
26	37		37 37			5.0	36		
27	97		97 97			5.0	96		
28	37		37 37			20.0	36		
29	97		97 97			20.0	96		
30		20		20	20	5.0			
31		60		60	60	5.0			
32		20		20	20	20.0			
33		60		60	60	20.0			

Case	NLN	ALN	DLN	IFAC	EFAC
34		10000.0		1	
35		10000.0		2	
36		10000.0		3	
37		10000.0		5	
38		10000.0		4	1000.0
39		10000.0		4	10.0

TABLE 4 Continued

Case	NLN	ALN	DLN	IFAC	ZFAC
40	2	1000.0, 1000.0	1.0, 1.0		
41	3	1000.0, 1000.0, 1000.0	1.0, 1.0, 1.0		

NOTE: Blanks indicate the basic value (Case 1, see Table 3).

TABLE 5

Input Parameters for Basic Double-Body Field
Used in Optimum Acceleration Parameter Studies

Case 42

IMAX	93			
JMAX	40			
NBDY	2			
IGES	0			
NBSEG	4			
NRSEG	2			
LBSID	1	1	1	3
LB1	1	75	29	1
LB2	19	93	65	93
LBDY	1	1	2	0
LRSID	1	2		
LR1	19	1		
LR2	29	40		
LISID	1	4		
LI1	65	1		
LI2	75	40		
R(1)	varied			
R(2)	0.0001			
R(3)	0.0001			
YINFIN	10.0			
AINFIN	-30.0			
XOINF	0.0			
YOINF	0.0			
NINF	92			

TABLE 5 Continued

IEV	0
R(10)	0.0
INFAC	0
INFACO	0
ATYP	ETA
ITYP	0
NLN	1
NPT	0
DEC	0.0
AMPFAC	0.0
JLN	1
ALN	1000.0
DLN	1.0
ATYP	0
ITYP	0
NLN	0
NPT	0
DEC	0.0
AMPFAC	0.0
IFAC	3
EFAC	100.0

TABLE 6

Input Parameters Varied for Double-Body Field Used in Optimum Acceleration Parameter Studies

Case	IMAX	JMAX	LB1	LB2	LR1	LR2	LI1	LI2	YINFIN	NINF	ALN	IFAC
45		60			19 1	29 60	65 1	75 60				
44		20			19 1	29 20	65 1	75 20				
47											10000.0	6
46											100.0	1
49									20.0			
48									5.0			
43	149		1 117	33 149	33 1	43 40	107 1	117 40		148		
			43 1	107 149								

Case	NLN	ALN	DLN	IFAC	EFAC
54		10000.0		6	10.0
53		10000.0		5	10.0
52		10000.0		5	
51		10000.0		4	
50		10000.0		3	

Note: Blanks indicate the basic value (Case 42, see Table 5).

TABLE 7

Optimum Acceleration Parameters for Single-Body Field:Variation of Single Quantity

Case Number	Case	Experimental Optimum		Variable Field	
		Optimum Acceleration Parameter	Number of Iterations	Average Variable Acceleration Parameter	Number of Iterations
	Effect of Number of Points on Airfoil				
4	IMAX = 37 (IMAX Down)	1.77	68	1.85	89
1	IMAX = 67 (IMAX Basic)	1.84	74	1.87	98
5	IMAX = 97 (IMAX Up)	1.88	137	1.90	> 137
	Effect of JMAX				
6	JMAX = 20 (JMAX Down)	1.81	90	1.83	> 90
1	JMAX = 40 (JMAX Basic)	1.84	74	1.87	98
7	JMAX = 60 (JMAX Up)	1.85	126	1.90	DIV
	Effect of Attraction Amplitude				
2	Amplitude = 100 (Amplitude Down)	1.85	81	1.88	97
1	Amplitude = 1000 (Amplitude Basic)	1.84	74	1.87	98
3	Amplitude = 10,000 (Amplitude Up)	1.86	114	1.86	> 114
	Effect of Outer Boundary Radius				
8	Radius = 5 (Radius Down)	1.80	205	1.91	DIV
1	Radius = 10 (Radius Basic)	1.84	74	1.87	98
9	Radius = 20 (Radius Up)	1.84	184	1.88	
	Effect of Body Shape				
1	Airfoil	1.84	74	1.87	98
0	Circle	1.88	96	1.87	> 96

TABLE 8

Optimum Acceleration Parameters for Single-Body Field : Variation of Pairs of Quantities

(12) IMAX Up Amplitude Down 1.89 : 122 (1.90 : >122)	(5) IMAX Up 1.88 : 137	(13) IMAX Up Amplitude Up 1.86 : 183 (1.89 : >183)
(2) Amplitude Down 1.85 : 81	(1) Basic 1.84 : 74	(3) Amplitude Up 1.86 : 114
(10) IMAX Down Amplitude Down 1.79 : 63 (1.85 : 92)	(4) IMAX Down 1.77 : 68	(11) IMAX Down Amplitude Up 1.83 : 86 (1.84 : 83)

(16) JMAX Up Amplitude Down 1.86 : 129 (1.90 : DIV)	(7) JMAX Up 1.85 : 126	(17) JMAX Up Amplitude Up 1.87 : 147 (1.90 : DIV)
(2) Amplitude Down 1.85 : 81	(1) Basic 1.84 : 74	(3) Amplitude Up 1.86 : 114
(14) JMAX Down Amplitude Down 1.84 : 89 (1.84 : 94)	(6) JMAX Down 1.81 : 90	(15) JMAX Down Amplitude Up 1.81 : 139 (1.81 : >139)

(20) Radius Up Amplitude Down 1.89 : 131 (1.88 : >131)	(9) Radius Up 1.84 : 205	(21) Radius Up Amplitude Up 1.81 : 243 (1.88 : DIV)
(2) Amplitude Down 1.85 : 81	(1) Basic 1.84 : 74	(3) Amplitude Up 1.86 : 114
(18) Radius Down Amplitude Down 1.80 : 185 (1.91 : DIV)	(8) Radius Down 1.80 : 184	(19) Radius Down Amplitude Up 1.81 : 176 (1.91 : DIV)

TABLE 8 Continued

(24) JMAX Up IMAX Down 1.83 : 77 (1.89 : >100)	(7) JMAX Up 1.85 : 126	(25) JMAX Up IMAX Up 1.89 : 128 (1.91 : DIV)
(4) IMAX Down 1.77 : 68	(1) Basic 1.84 : 74	(5) IMAX Up 1.88 : 137
(22) JMAX Down IMAX Down 1.71 : 48 (1.76 : 62)	(6) JMAX Down 1.81 : 90	(23) JMAX Down IMAX Up 1.84 : 142 (1.87 : >142)

(28) Radius Up IMAX Down 1.81 : 77 (1.84 : 92)	(9) Radius Up 1.84 : 205	(29) Radius Up IMAX Up 1.87 : 219 (1.90 : >219)
(4) IMAX Down 1.77 : 68	(1) Basic 1.84 : 74	(5) IMAX Up 1.88 : 137
(26) Radius Down IMAX Down 1.78 : 102 (1.84 : >102)	(8) Radius Down 1.80 : 184	(27) Radius Down IMAX Up 1.82 : 285 (1.93 : DIV)

(32) Radius Up JMAX Down 1.82 : 117 (1.84 : >117)	(9) Radius Up 1.84 : 205	(33) Radius Up JMAX Up 1.89 : 103 (1.90 : >103)
(6) JMAX Down 1.81 : 90	(1) Basic 1.84 : 74	(7) JMAX Up 1.85 : 126
(30) Radius Down JMAX Down 1.80 : 176 (1.91 : DIV)	(8) Radius Down 1.80 : 184	(31) Radius Down JMAX Up 1.58 : 218 (1.91 : DIV)

TABLE 8 Continued

Format: (Case Number) Case
Optimum Acceleration Parameter: Number of Iterations
(Average Variable Acceleration Parameter: Number of Iterations)

See Tables 3-4 for values of the quantities varied.

TABLE 9

Optimum Acceleration Parameter
for Single-Body Field : Miscellaneous Variations

Case Number	Case	Experimental Optimum	
		Optimum Acceleration Parameter	Number of Iterations
	Effect of Number of Steps in Addition of Inhomogeneous Term (Attraction Up)		
34	One Step	1.35	561
35	Two Step	1.61	317
36	Three Step	1.85	119
3	Four Step	1.86	114
37	Five Step	1.85	125
	Effect of Intermediate Convergence Criterion (Attraction Up)		
38	Convergence : -01	1.86	101
3	Convergence : -02	1.86	114
39	Convergence : -03	1.85	167
	Effect of Number of Attraction Lines (Basic)		
1	One Line	1.84	74
40	Two Lines	1.83	76
41	Three Lines	1.83	79
	Effect of Convergence Criterion (Basic)		
1	Convergence : -03	1.84	49
1	Convergence : -04	1.84	74
1	Convergence : -05	1.84	117
1	Convergence : -06	1.89	175

166

TABLE 10

Optimum Acceleration Parameters for Two-Body Field

Case Number	Case	Experimental Optimum	
		Optimum Acceleration Parameter	Number of Iterations
	Effect of Number of Points on Airfoils		
42	37 Points (IMAX Basic)	1.84	189
43	65 Points (IMAX Up)	1.86	304
	Effect of JMAX		
44	JMAX = 20 (JMAX Down)	1.78	203
42	JMAX = 40 (JMAX Basic)	1.84	189
45	JMAX = 60 (JMAX Up)	1.85	238
	Effect of Attraction Amplitude		
46	Amplitude = 100 (Amplitude Down)	1.85	176
42	Amplitude = 1000 (Amplitude Basic)	1.84	189
47	Amplitude = 10,000 (Amplitude Up)	1.59	477
	Effect of Outer Boundary Radius		
48	Radius = 5 (Radius Down)	1.61	285
42	Radius = 10 (Radius Basic)	1.84	189
49	Radius = 20 (Radius Up)	1.86	263
	Effect of Number of Steps in Addition of Inhomogeneous Term (Attraction Up)		
50	Three Steps	Divergence	
51	Four Steps	1.59	466
52	Five Steps	1.59	472
47	Six Steps	1.59	477

TABLE 10 Continued

Case Number	Case	Experimental Optimum	
		Optimum Acceleration Parameter	Number of Iterations
	Effect of Intermediate Convergence Criterion (Attraction Up)		
(52)	Five Steps, Convergence : -02	1.59	472
(53)	Five Steps, Convergence : -03	1.63	570
(47)	Six Steps, Convergence : -02	1.59	477
(54)	Six Steps, Convergence : -03	1.63	602

TABLE 11

Input Parameters for Coordinate System Control Demonstration of Figure 27

IMAX = 31, JMAX = 30, YINFIN = 10, AINFIN = 0.0

(a.) No Attraction

(b.) ETA Attraction: JLN = 20, ALN = 10.0, DLN = 1.0

(c.) ETA Attraction: IPT = 1, JPT = 20, APT = 10.0, DPT = 1.0
IPT = 16, JPT = 20, APT = 10.0, DPT = 1.0

(d.) ETA Attraction: IPT = 5, JPT = 2, APT = 1000.0, DPT = 1.0
IPT = 6, JPT = 2, APT = 1000.0, DPT = 100.0

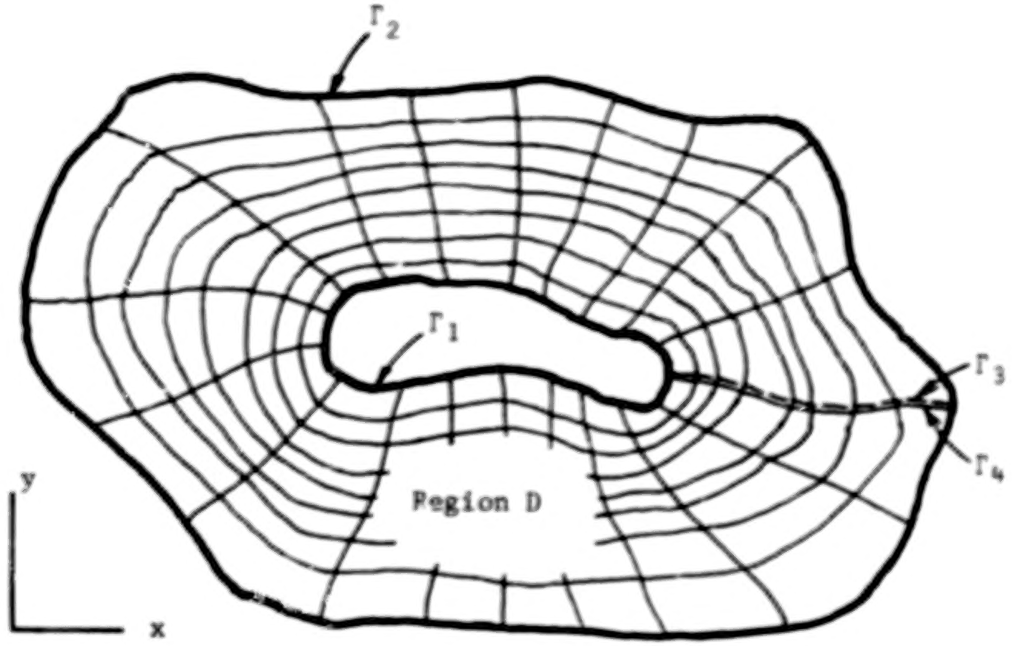
(e.) XI Attraction: JLN = 2, ALN = 100.0, DLN = 1.0

(f.) XI Attraction: IPT = 5, JPT = 1, APT = 1000.0, DPT = 1.0

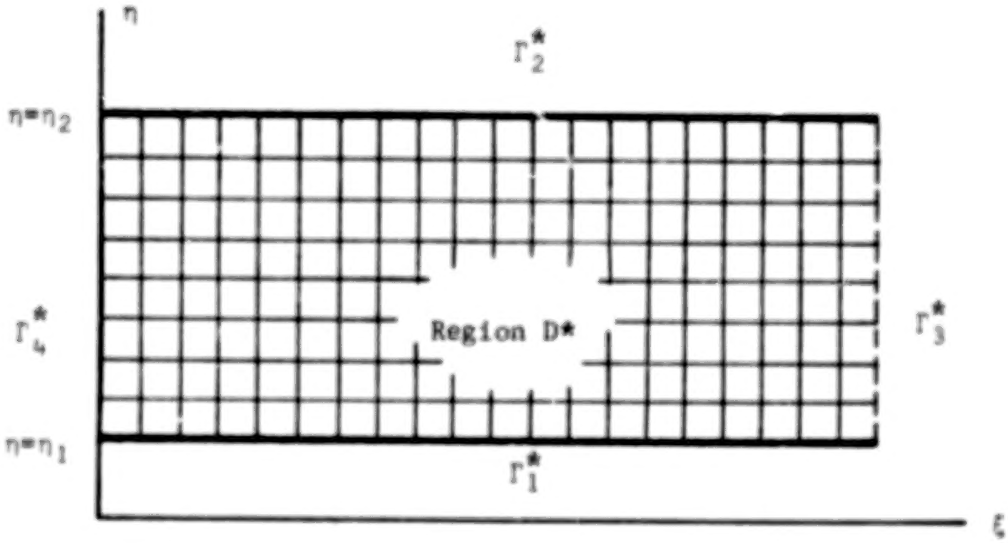
Table 12. Coordinate System Control Parameters for Figures 33, 34, 37, and 38.

169,

Figure 33.	Coordinate line attraction to the first 15 lines around the airfoil with amplitude 20,000 on the body, varying linearly to 13,000 on the 15th line. Decay factor of 1.0 for all except last line, where 0.4 was used. 75 points on airfoil, 58 lines around airfoil. Circular boundary of radius 10 chords.
Figure 34.	Coordinate line attraction to the first 15 lines around the airfoil with amplitude 20,000 on the body, varying linearly to 13,000 on the 15th line. Decay factor of 1.0 for all except last line, where 0.4 was used. 75 points on airfoil, 58 lines around airfoil. Circular boundary of radius 10 chords.
Figure 37.	Coordinate line attraction to the entire boundary with amplitude 100 and decay factor 0.1.
Figure 38.	Coordinate line attraction to the first 15 lines around the body with amplitude 20,000 on the body varying linearly to 13,000 on the 15th line. Decay factor of 1.0 for all except last line where 0.4 was used. 75 points on body, 58 lines around body. Circular boundary of radius 10 chords.

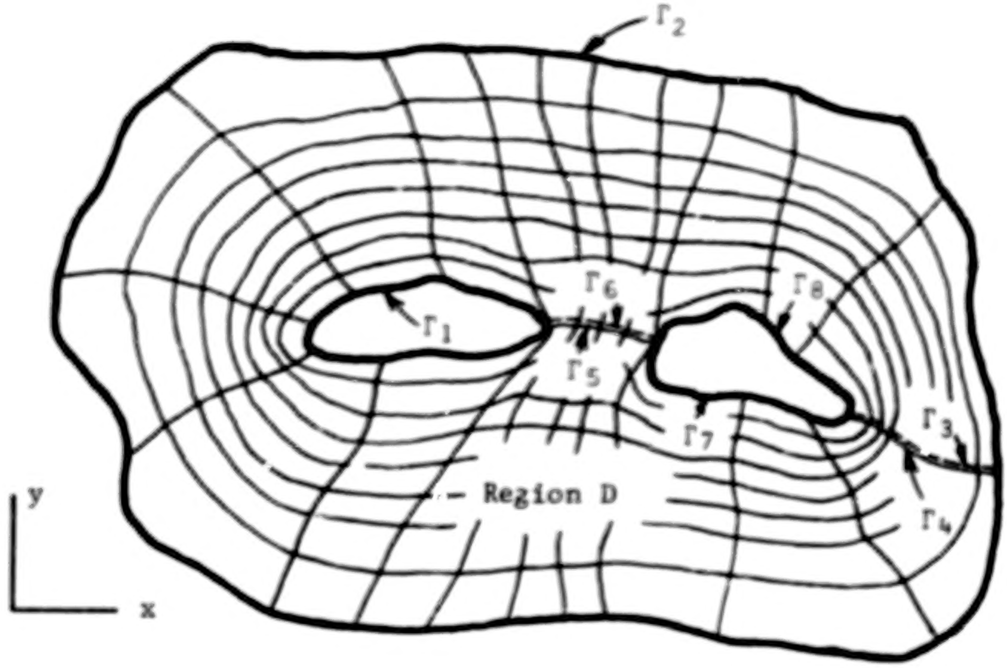


Physical Plane

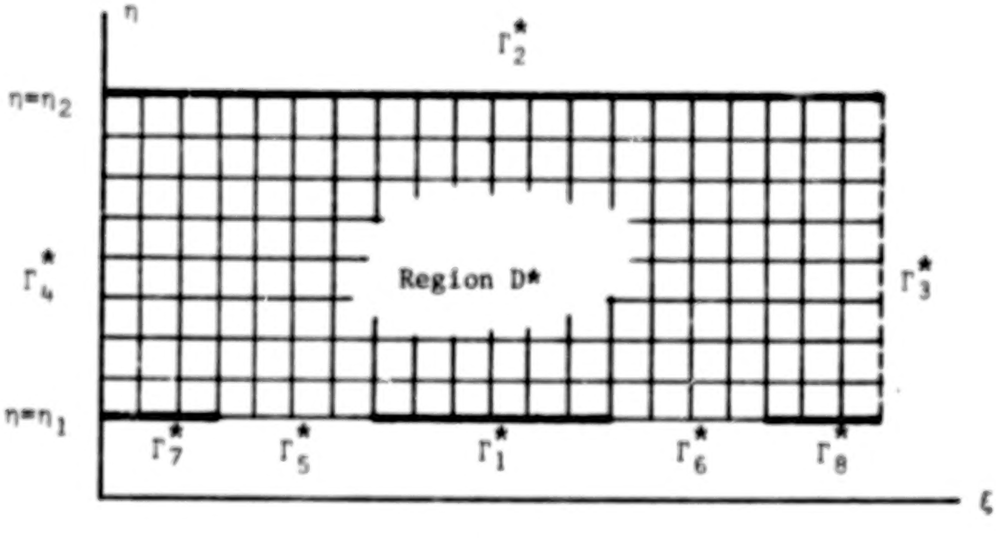


Transformed Plane

Figure 1. Field Transformation - Single Body

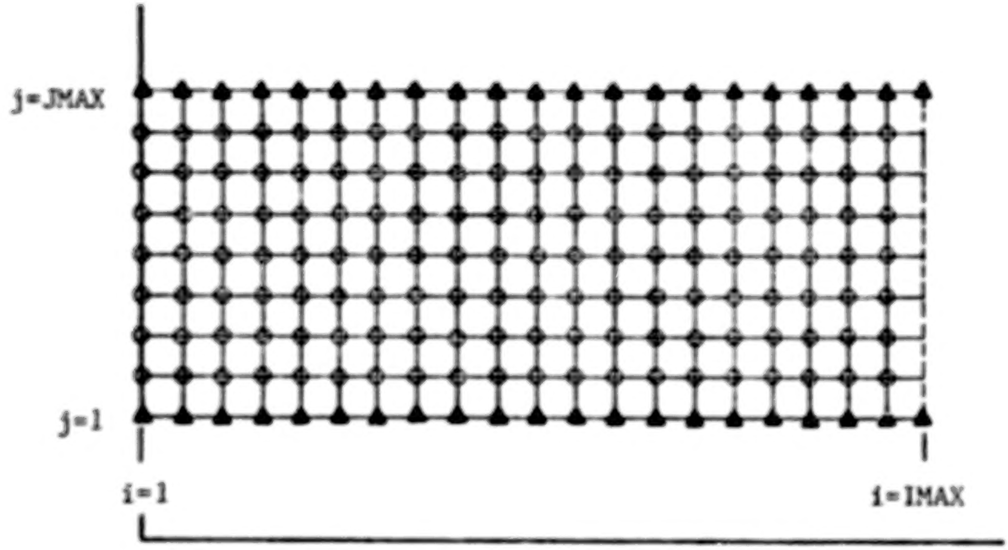


Physical Plane

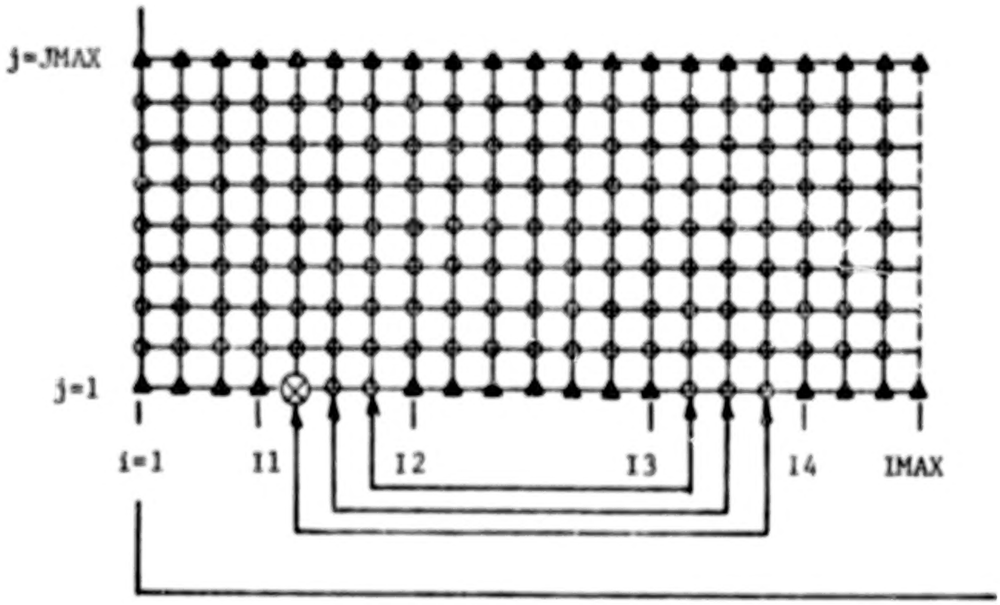


Transformed Plane

Figure 2. Field Transformation - Multiple Bodies



a) Single Body



b) Two Body

Figure 3. Computational Grids - Single and Two Body Regions

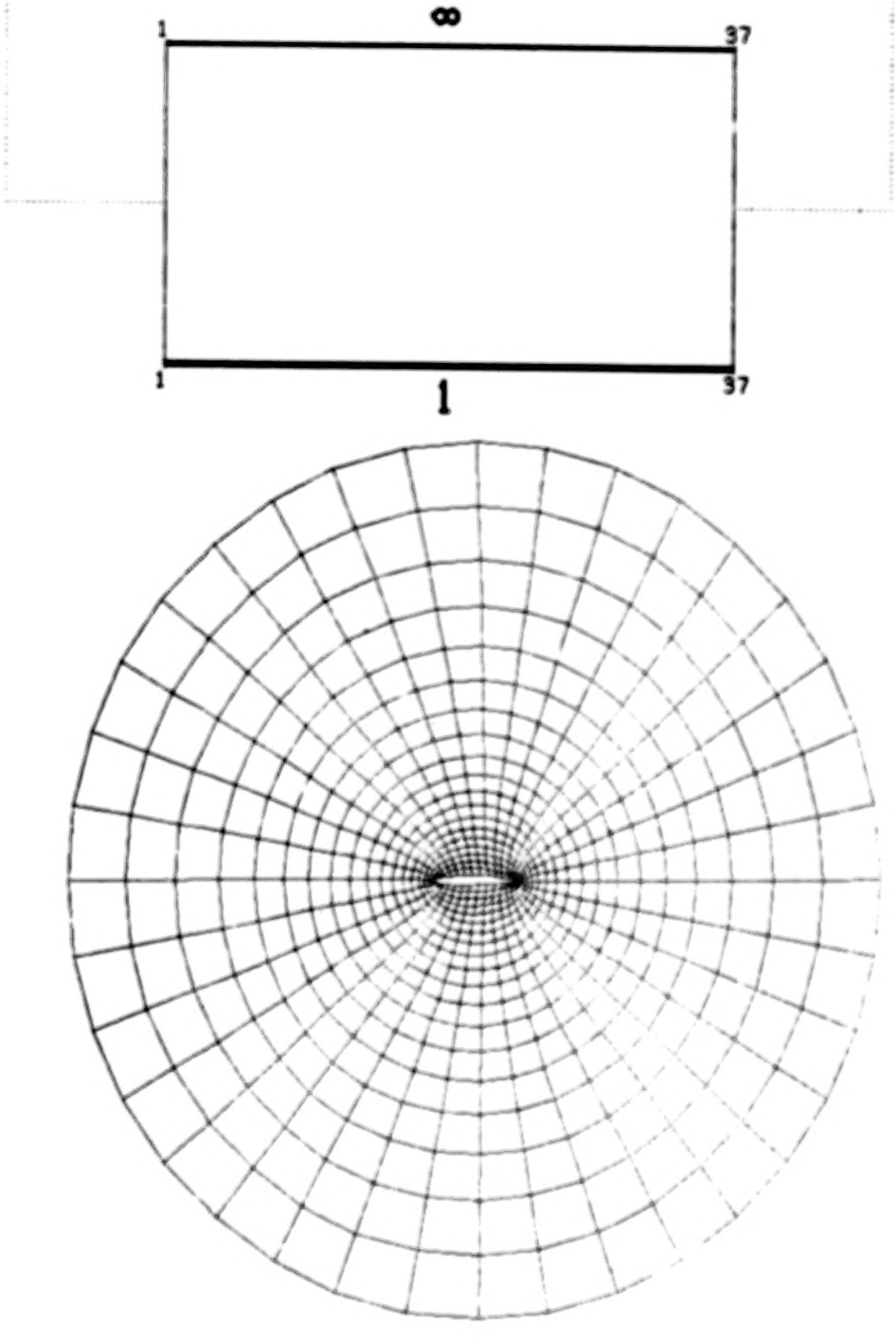
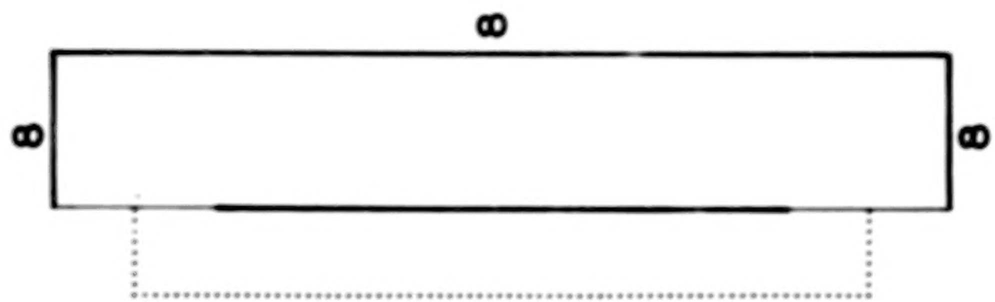
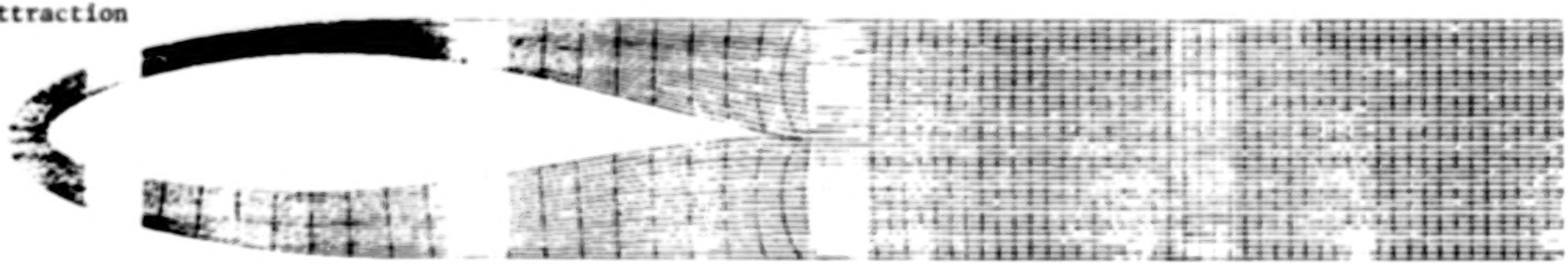


Figure 4a. Single-Body Configuration

174



No Coordinate
Attraction



With Coordinate
Attraction

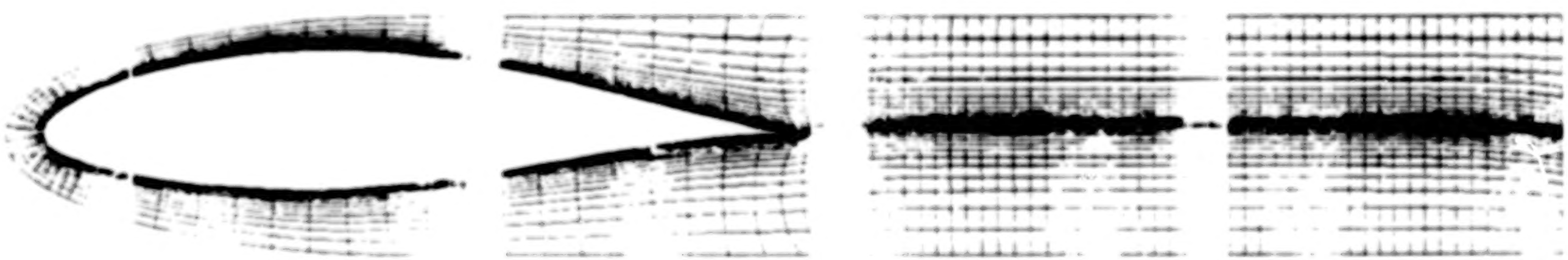


Figure 4b. /lternate Single-Body Configuration

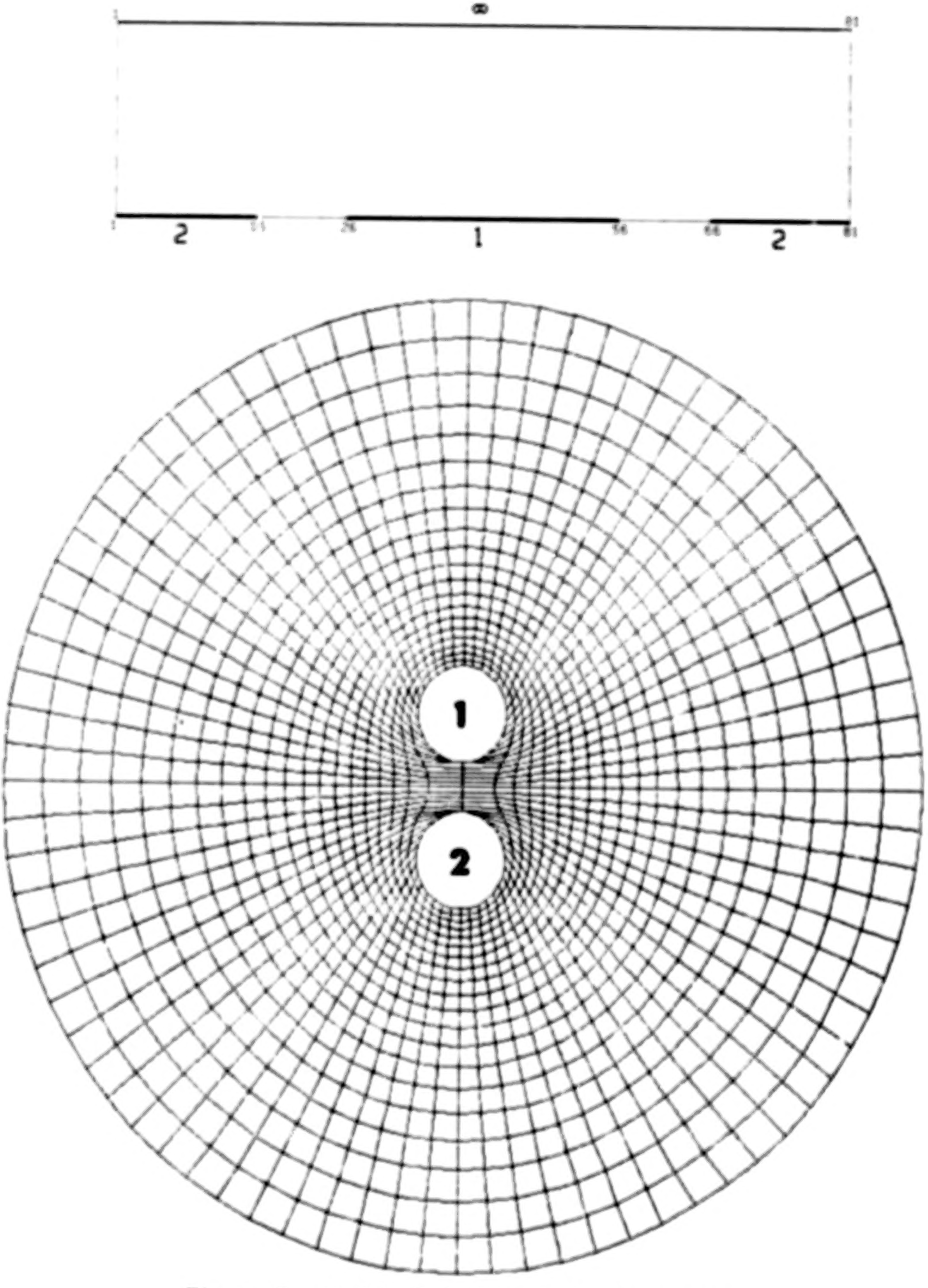


Figure 5. Double-Body Segment Configuration #1

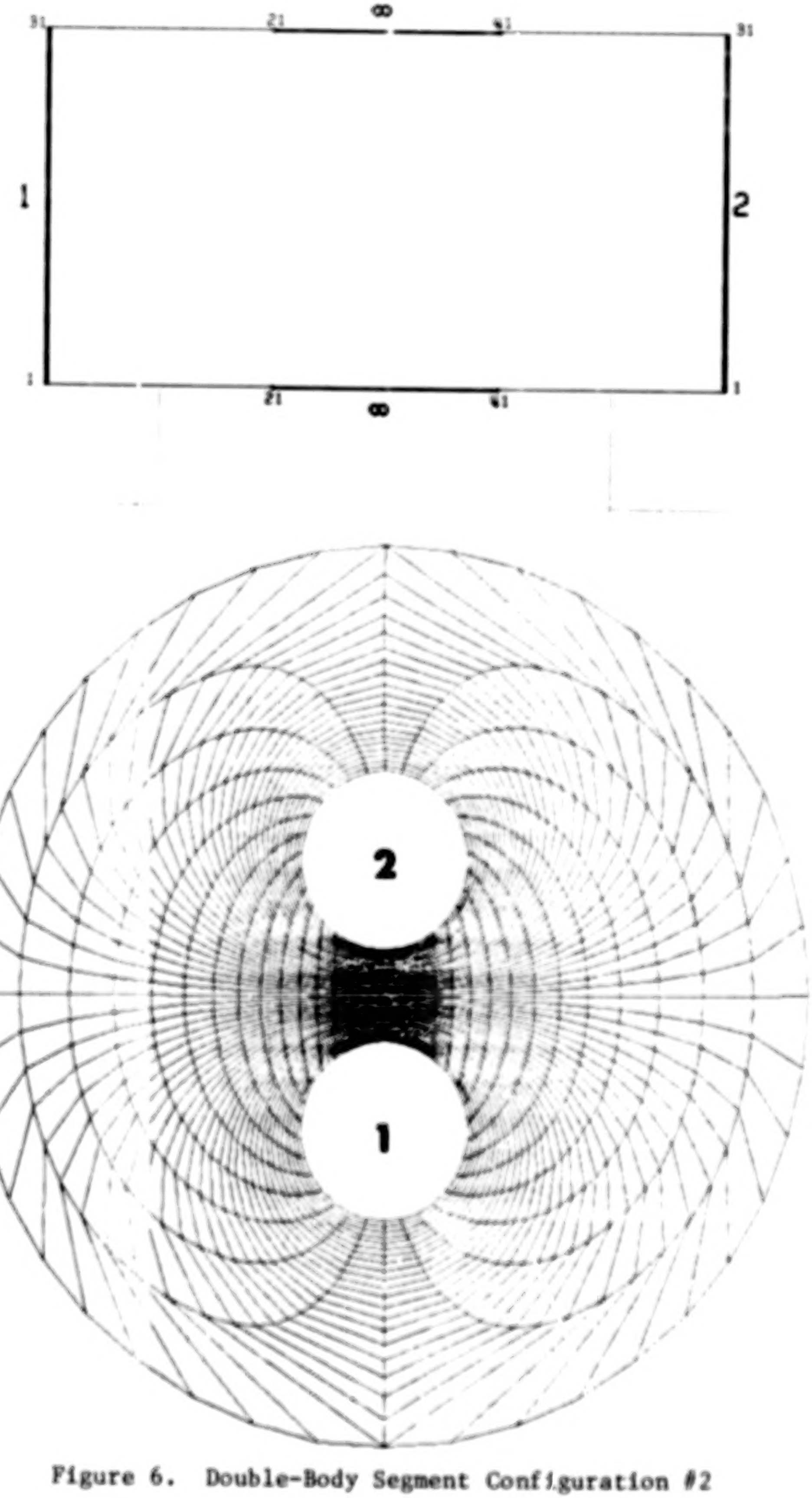


Figure 6. Double-Body Segment Configuration #2

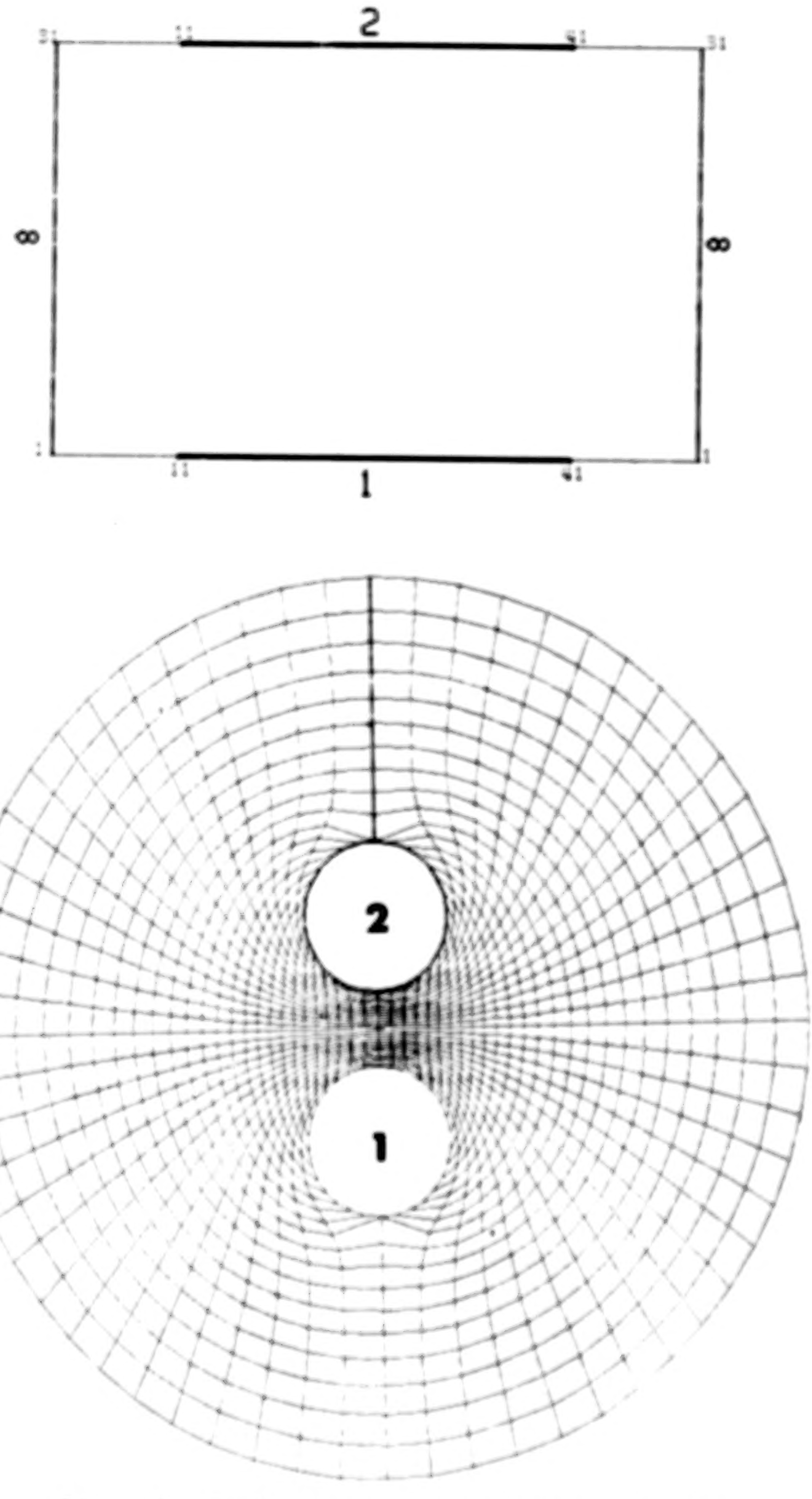


Figure 7. Double-Body Segment Configuration #3

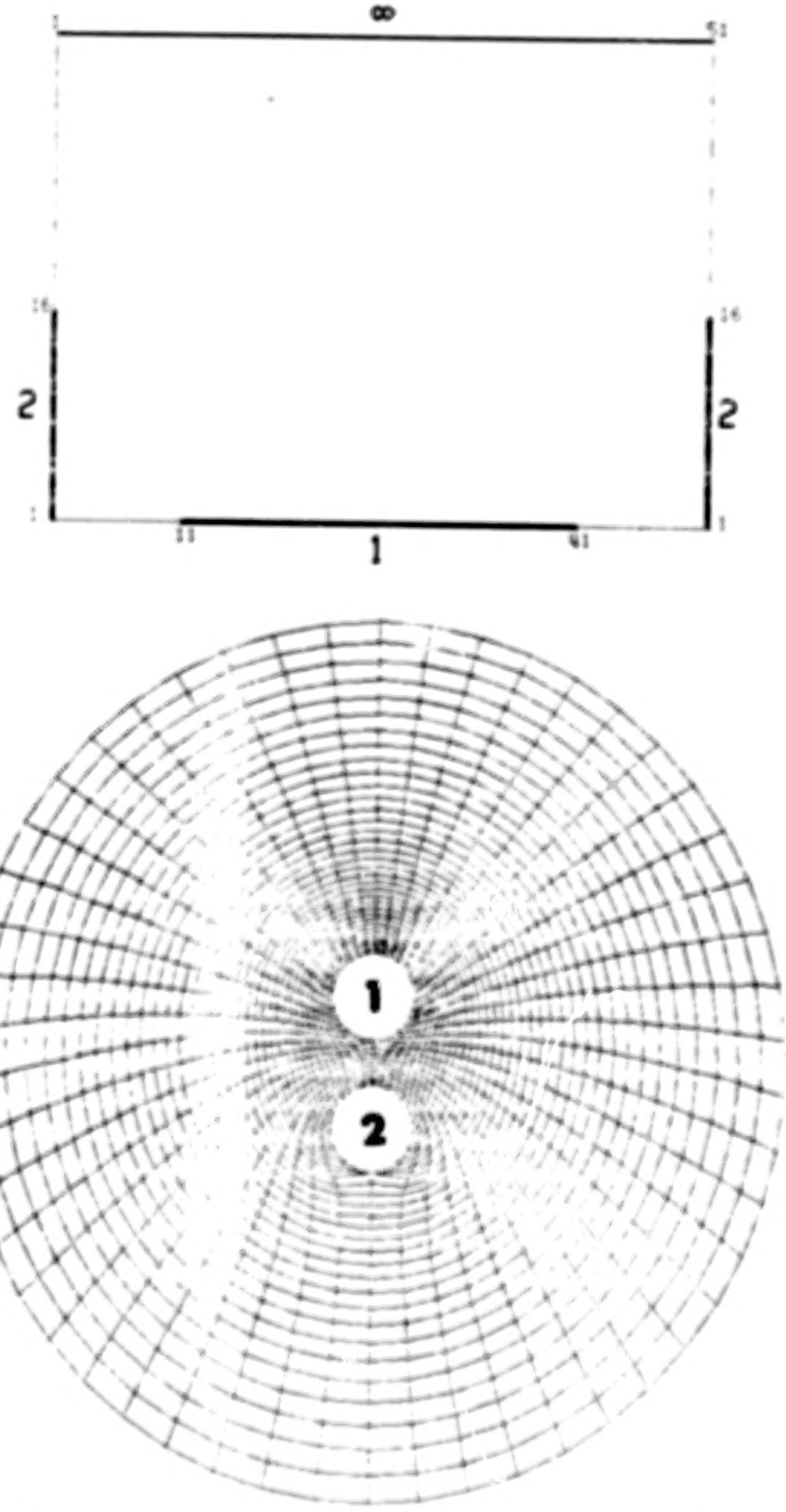
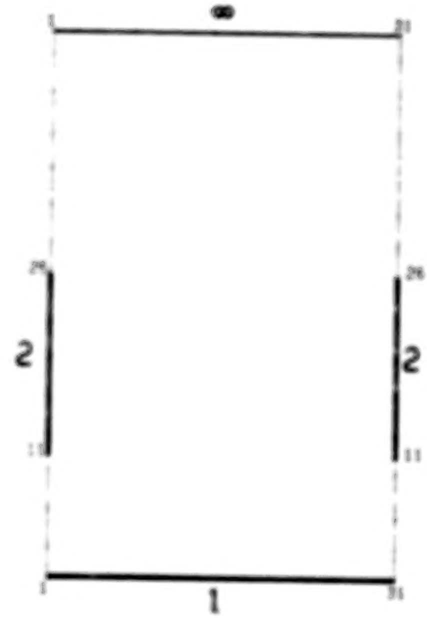


Figure 8. Double-Body Segment Configuration #4



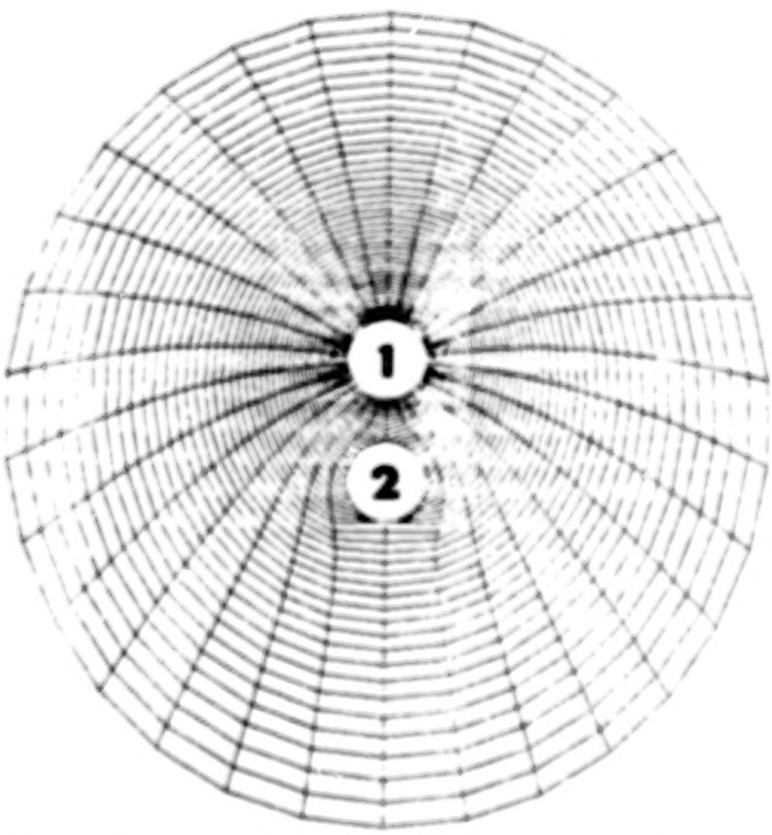


Figure 9. Double-Body Segment Configuration #5

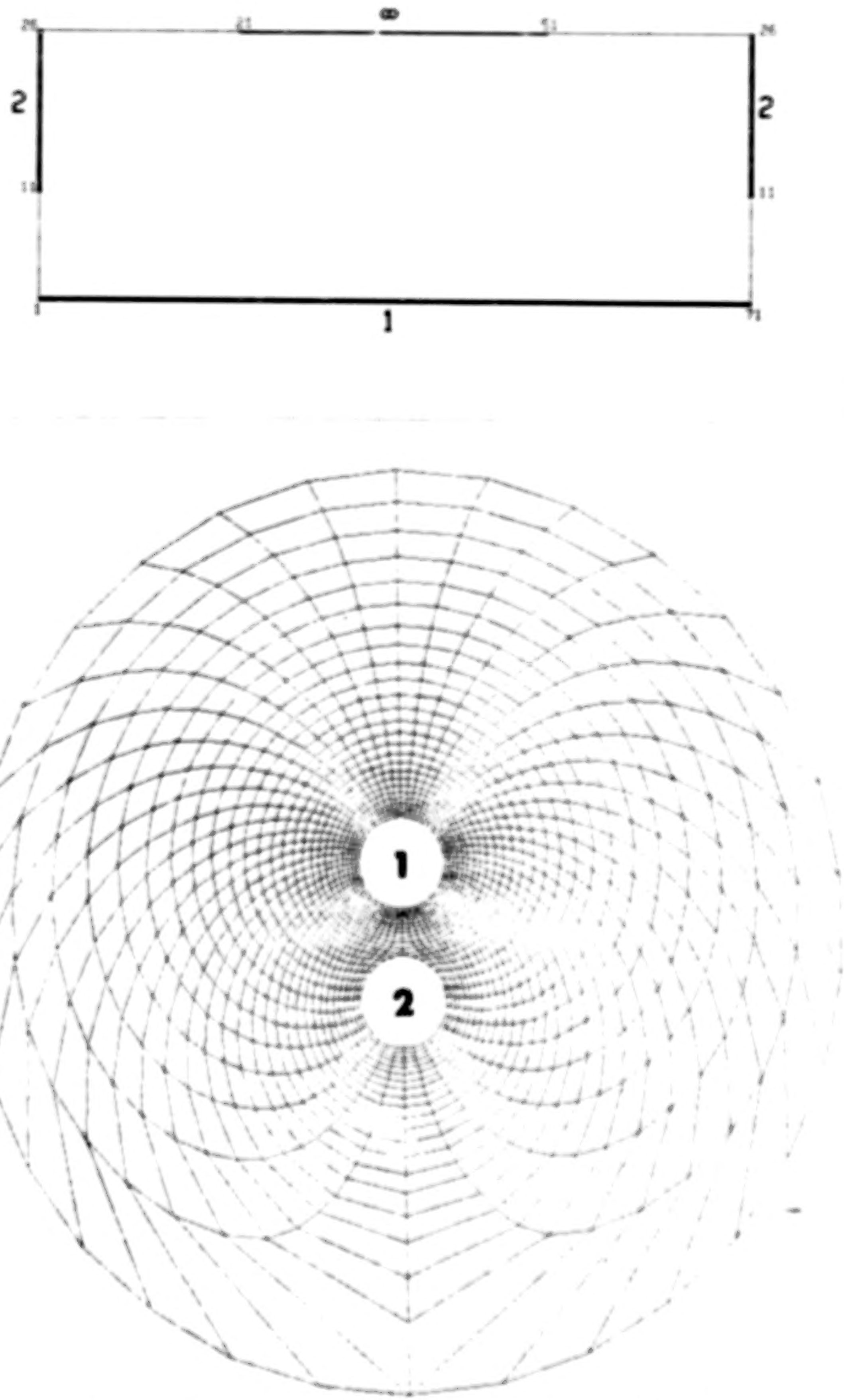


Figure 10. Double-Body Segment Configuration #6

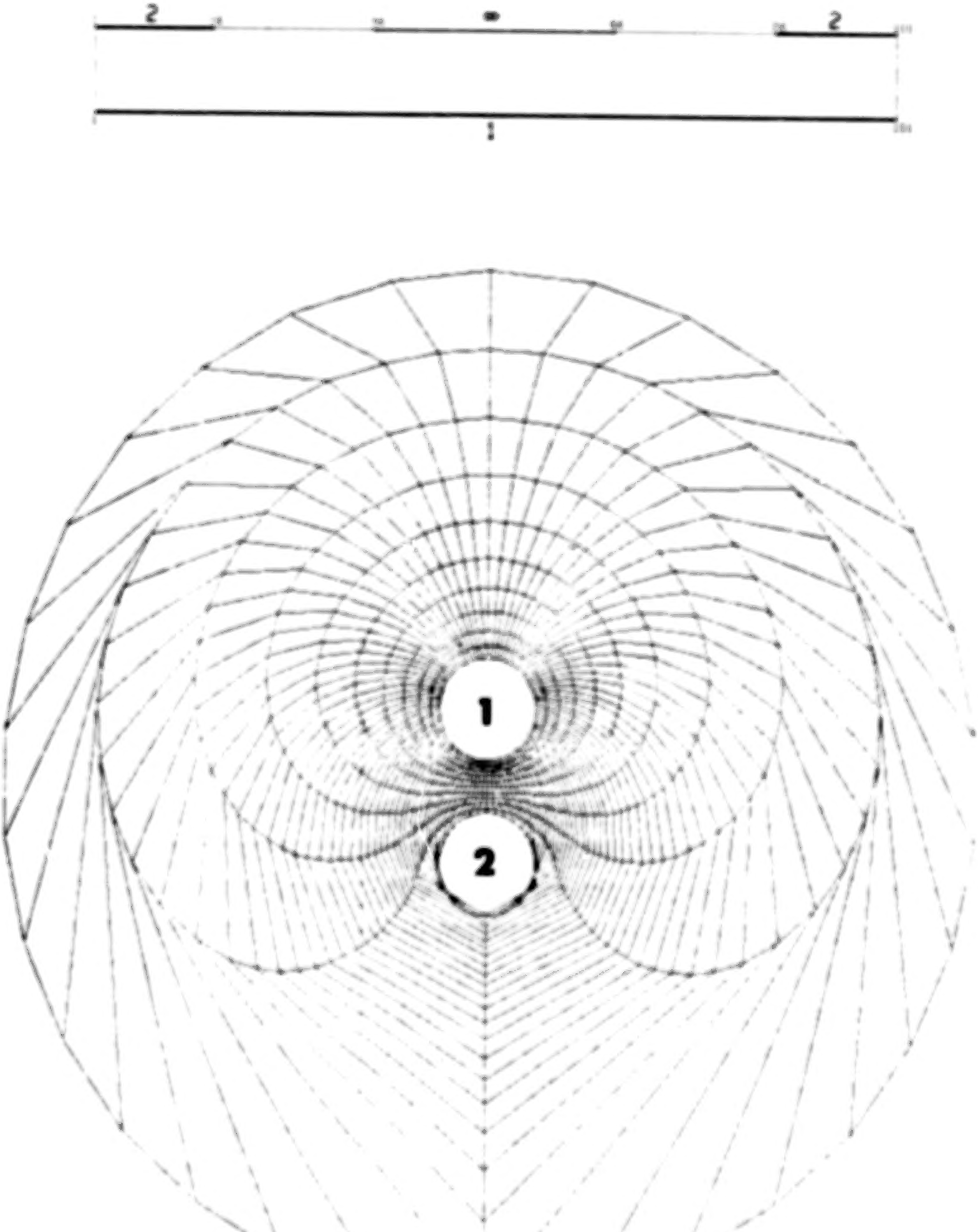


Figure 11. Double-Body Segment Configuration #7

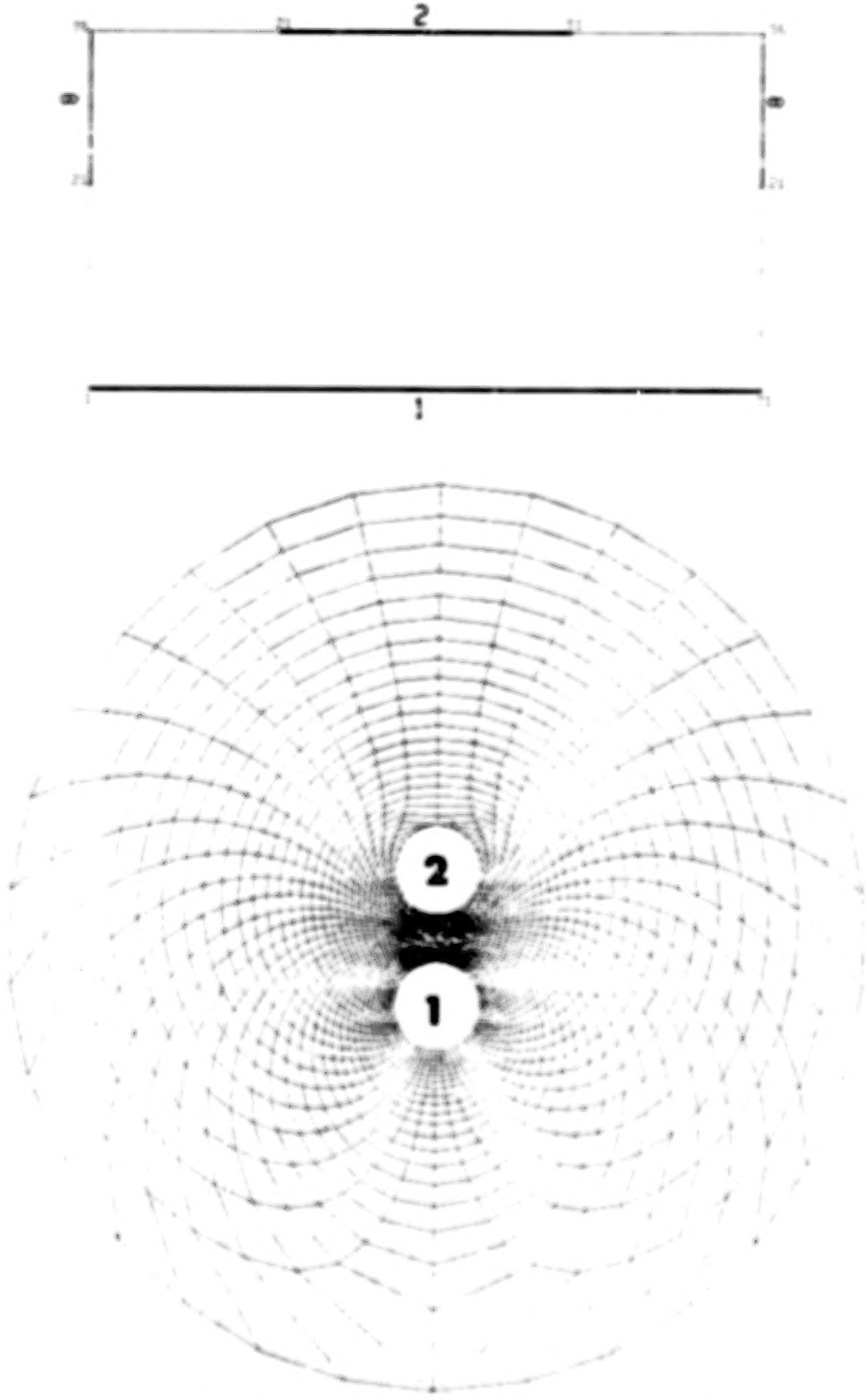


Figure 12. Double-Body Segment Configuration #8

TABLE OF CONTENTS

TABLE OF CONTENTS, CONTINUED

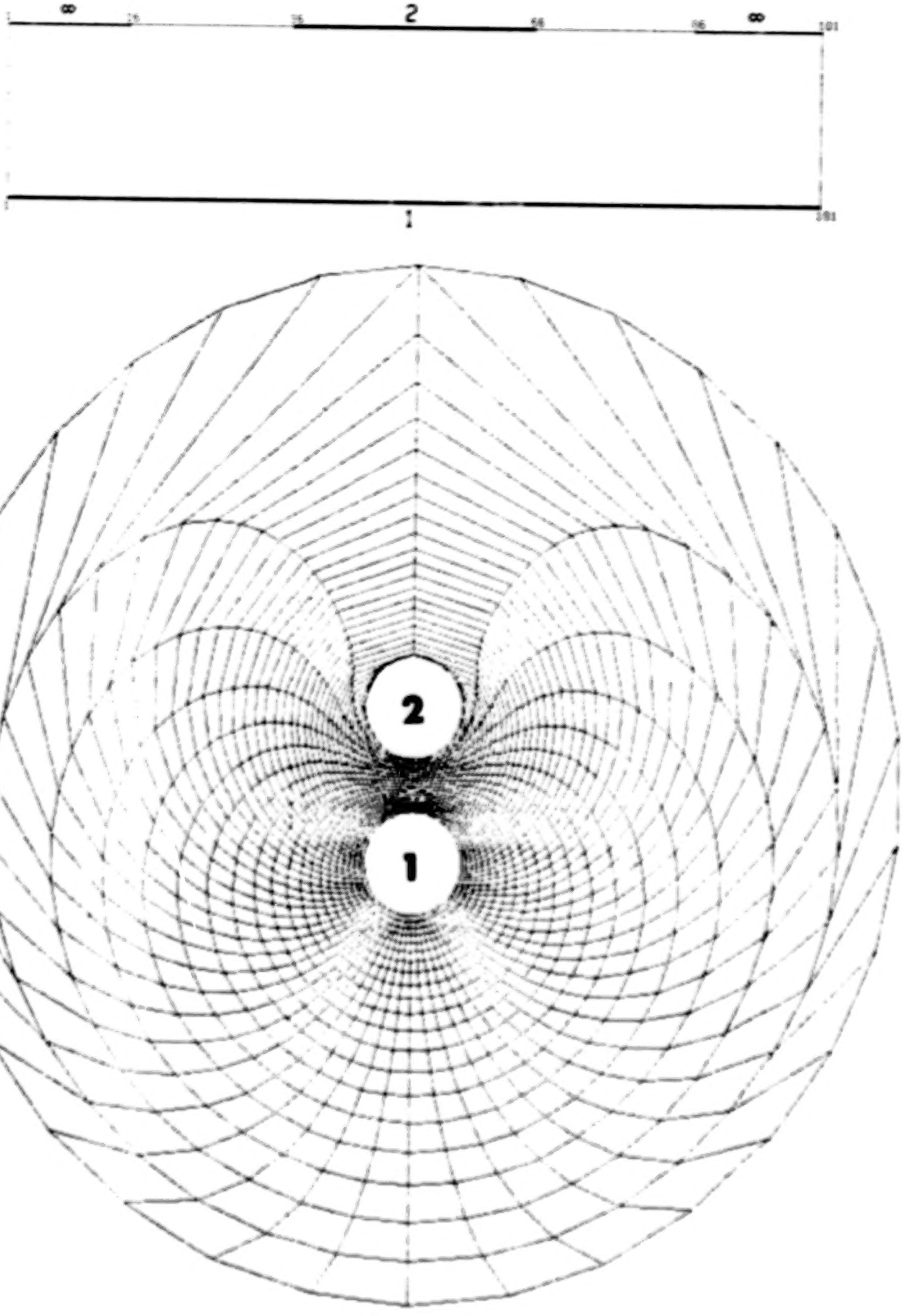


Figure 13. Double-Body Segment Configuration #9

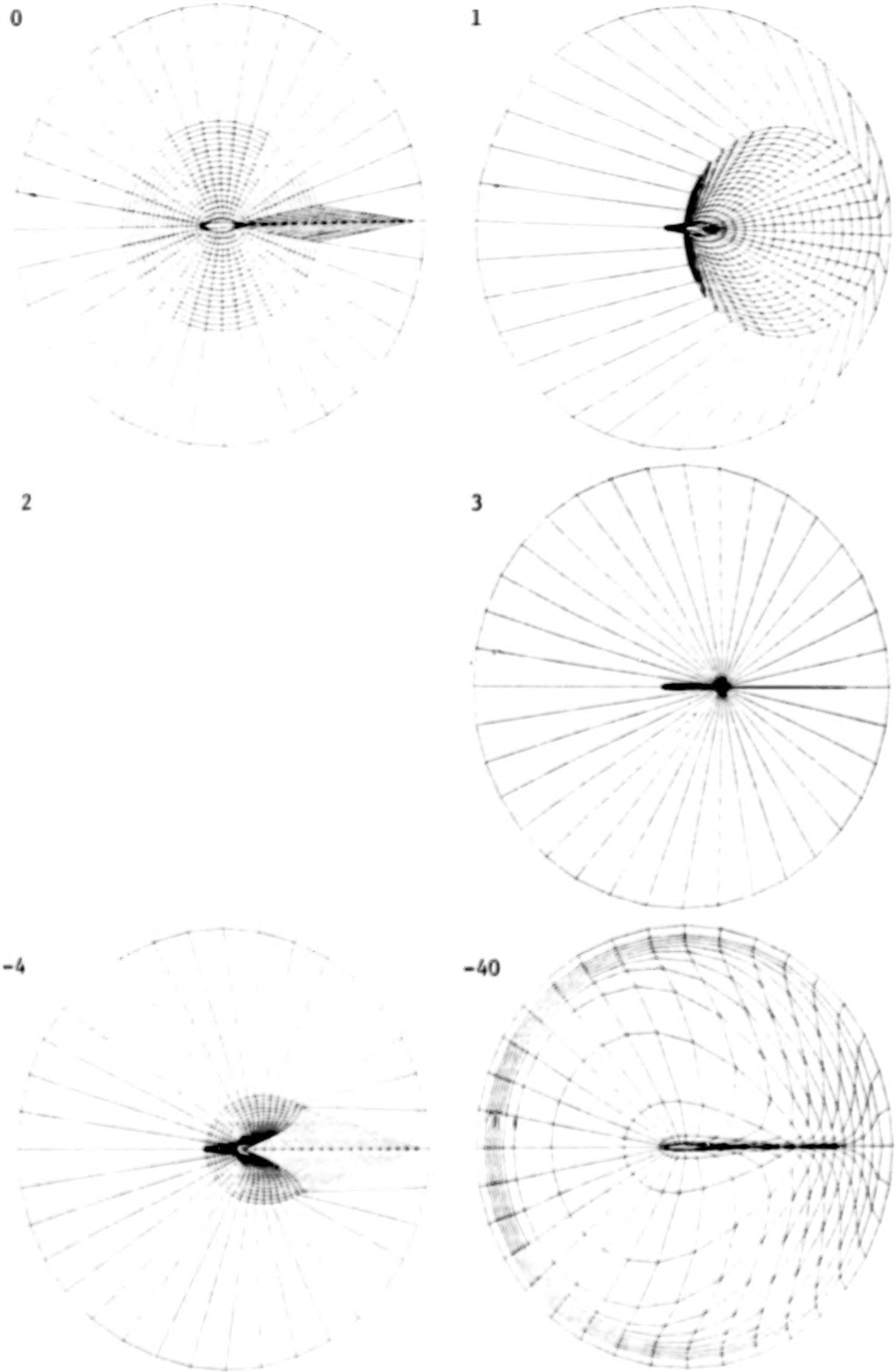


Figure 14. Initial Guess Types - Single Body

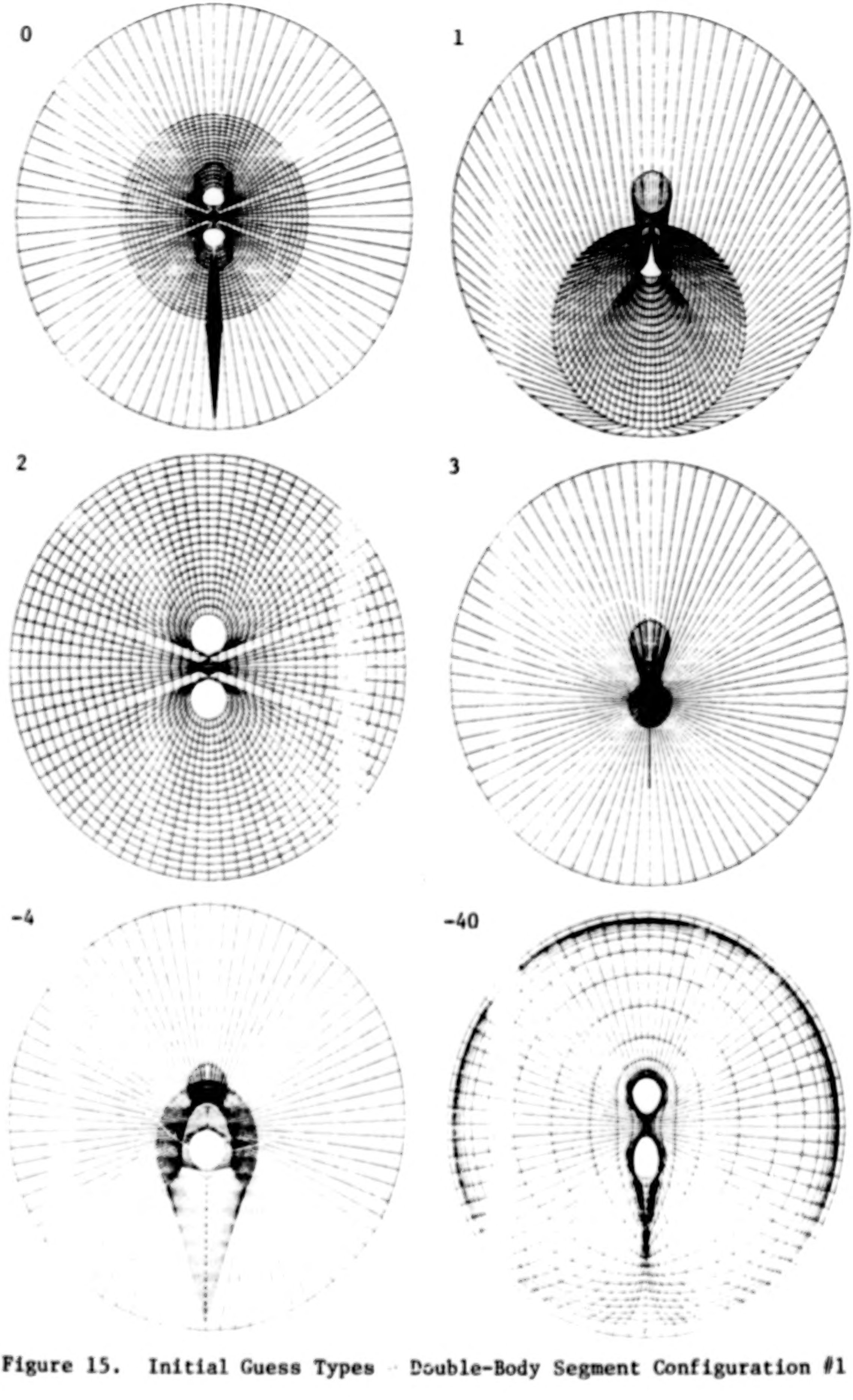


Figure 15. Initial Guess Types - Double-Body Segment Configuration #1

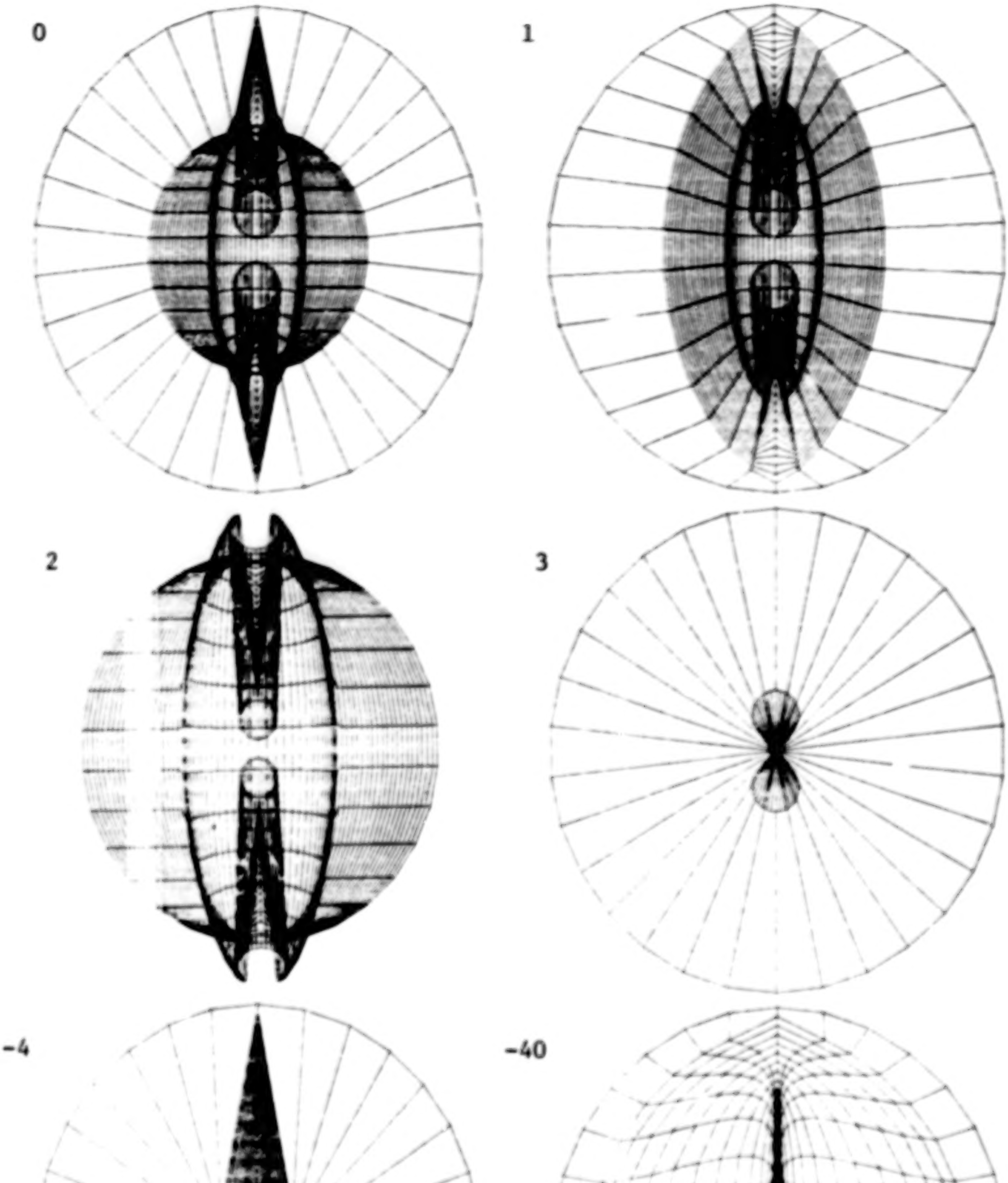


Figure 16. Initial Guess Types - Double-Body Segment Configuration #2

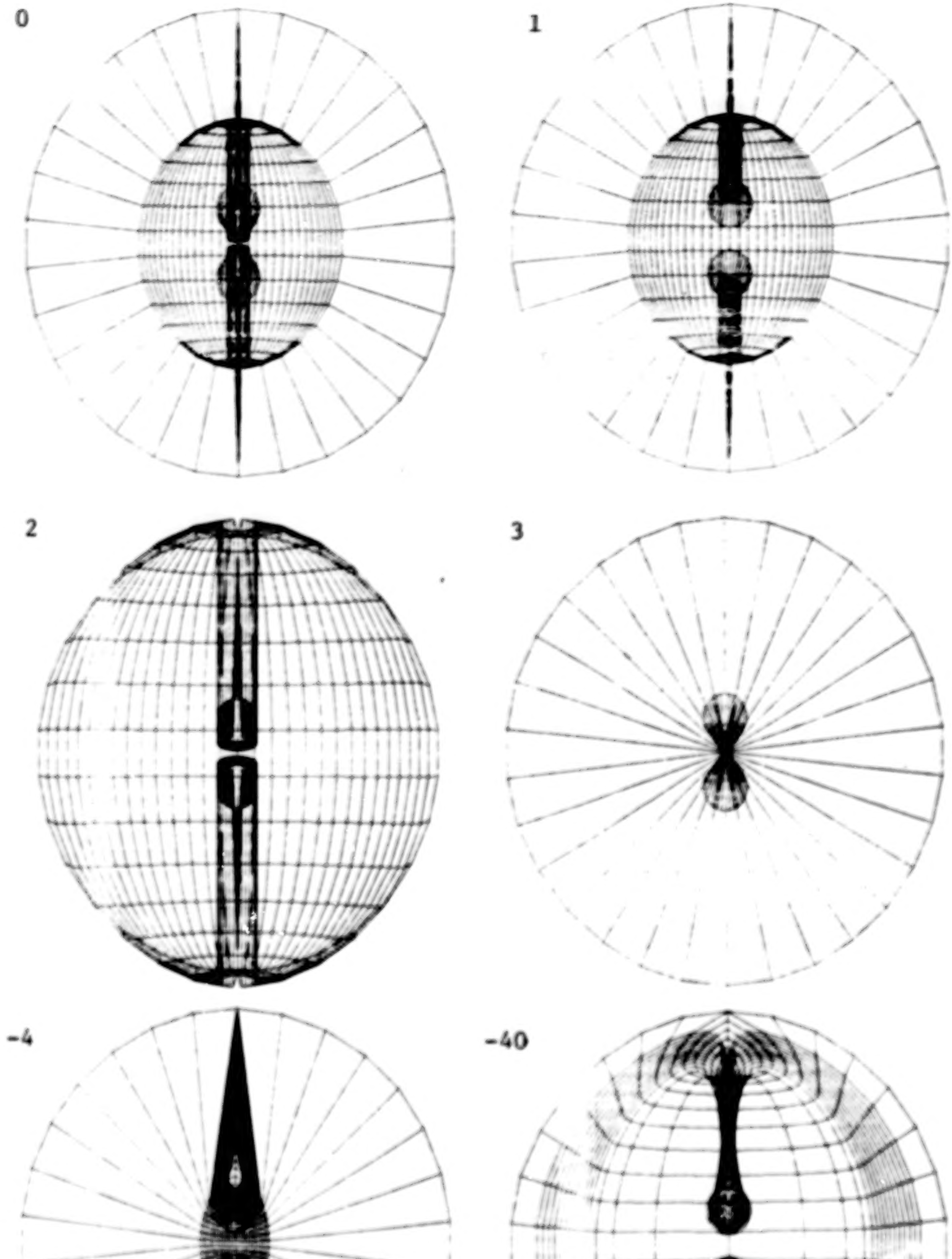


Figure 17. Initial Guess Types - Double-Body Segment Configuration #3

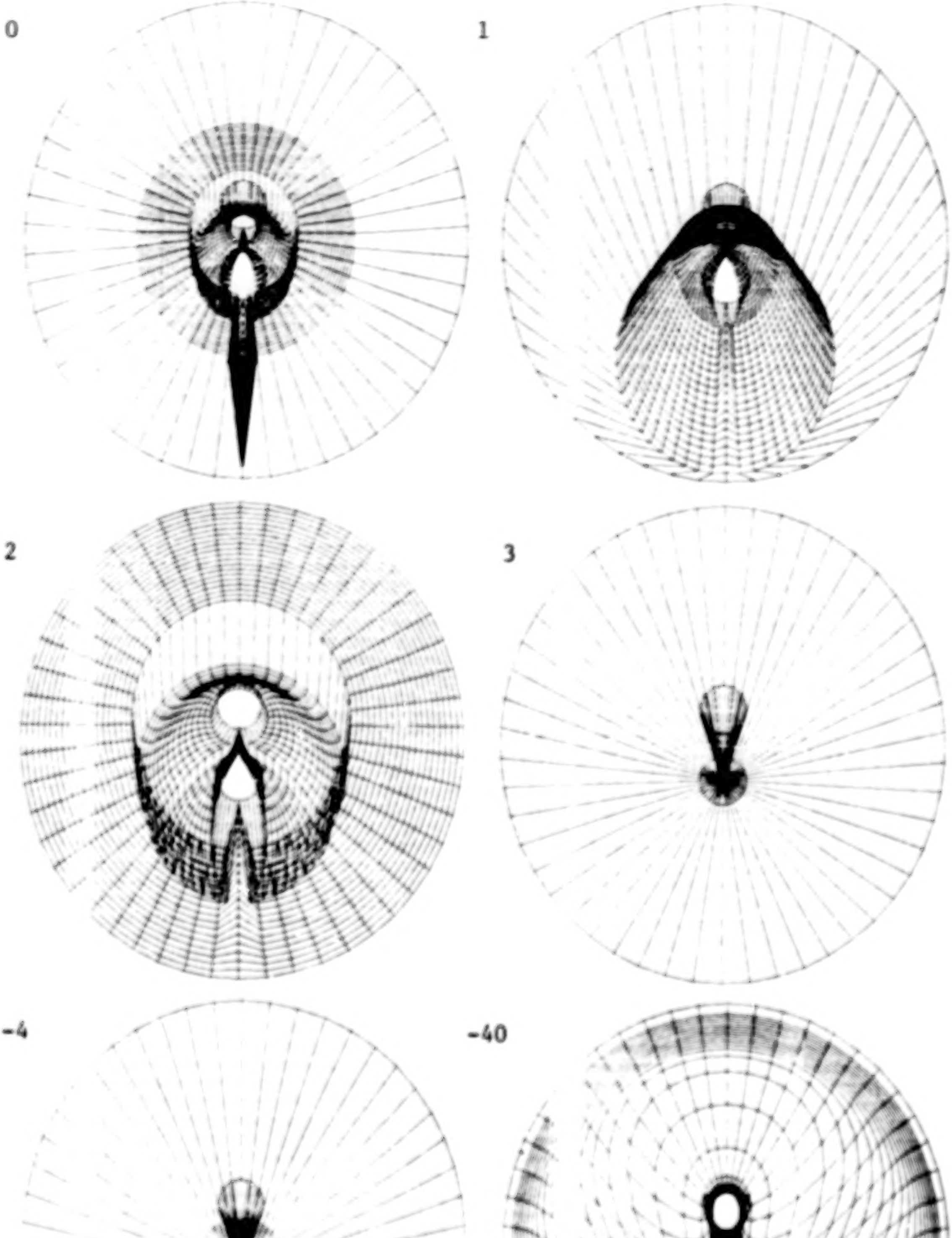


Figure 18. Initial Guess Types - Double-Body Segment Configuration #4

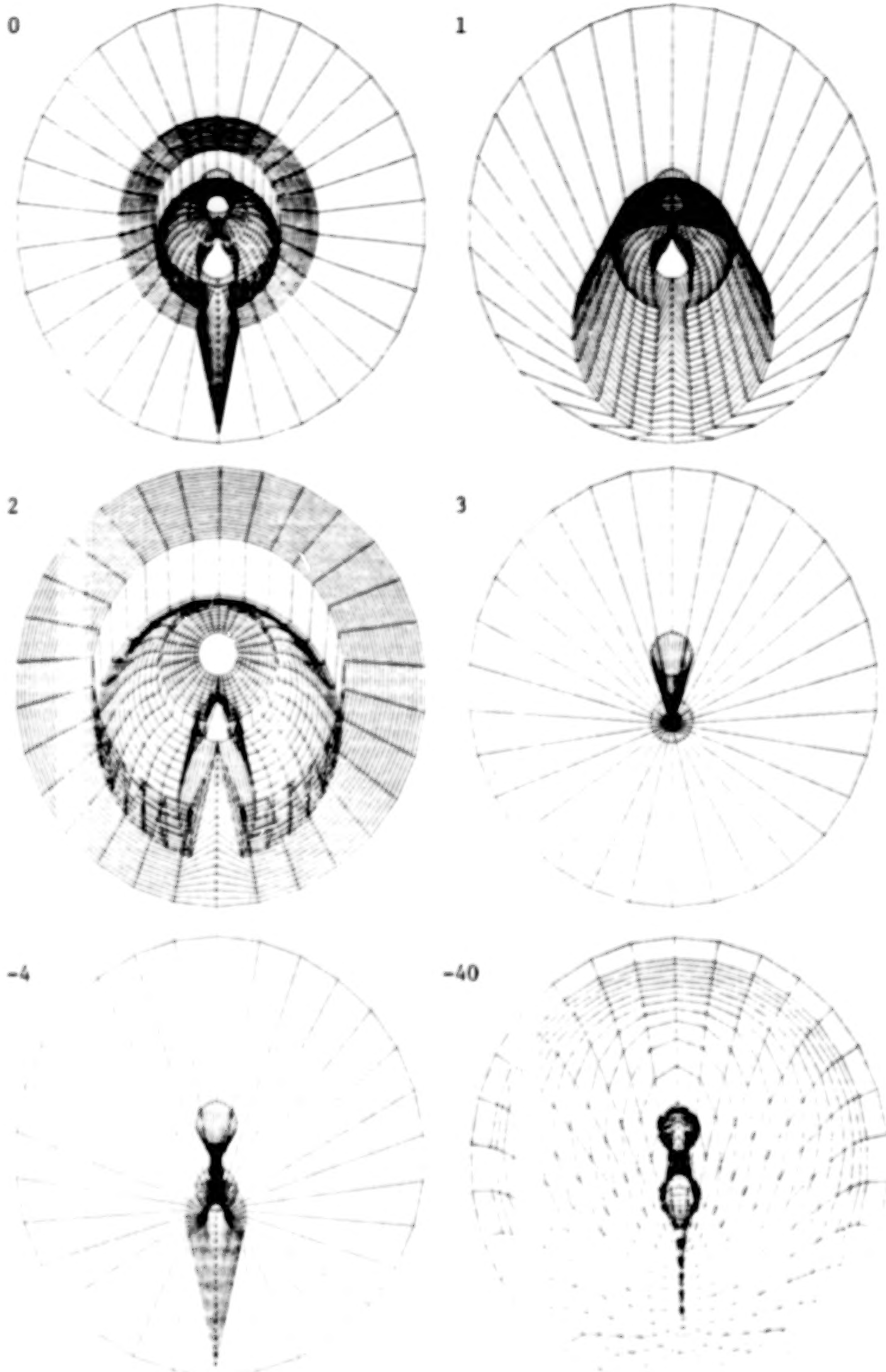


Figure 19. Initial Guess Types - Double-Body Segment Configuration #5

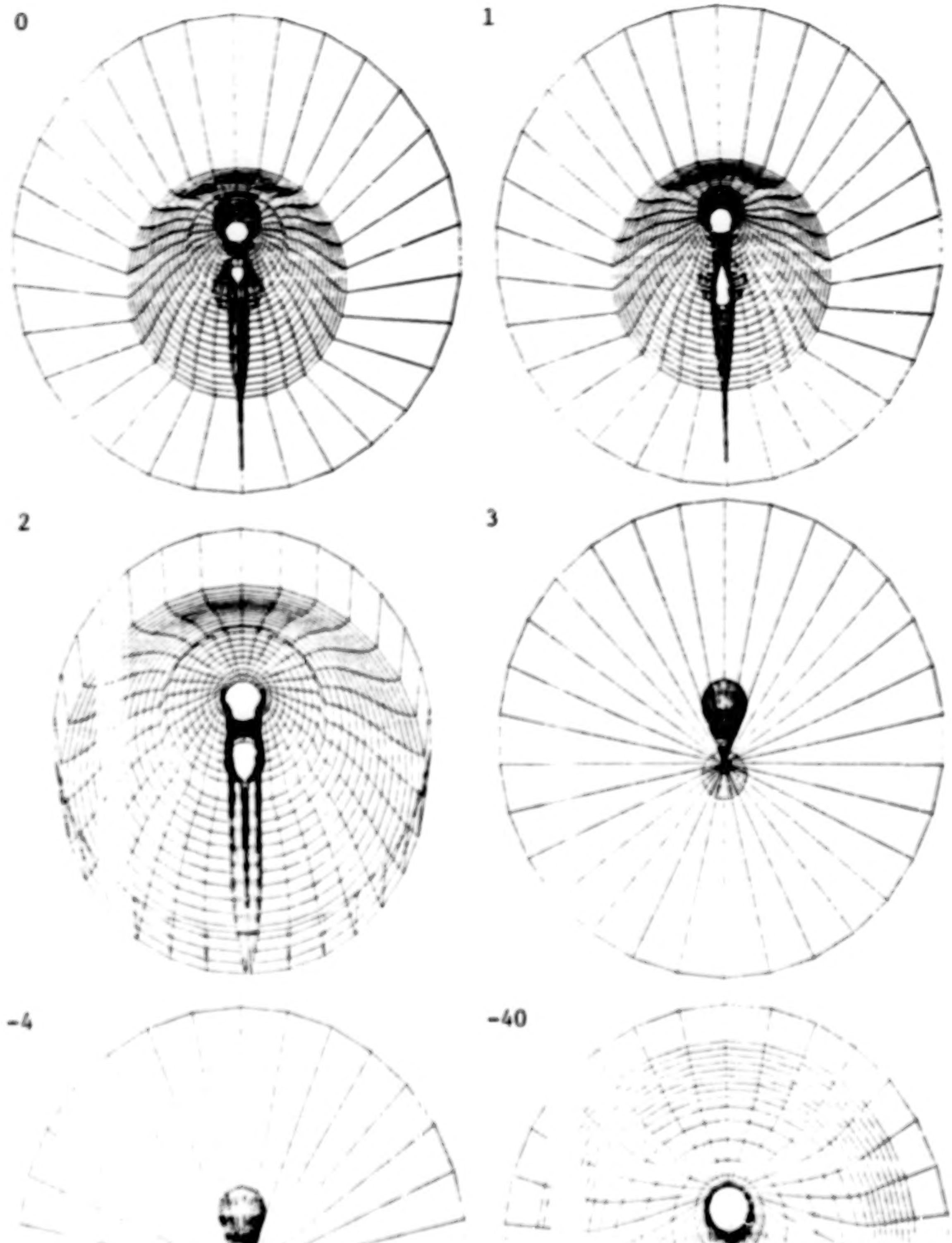


Figure 20. Initial Guess Types - Double-Body Segment Configuration #6

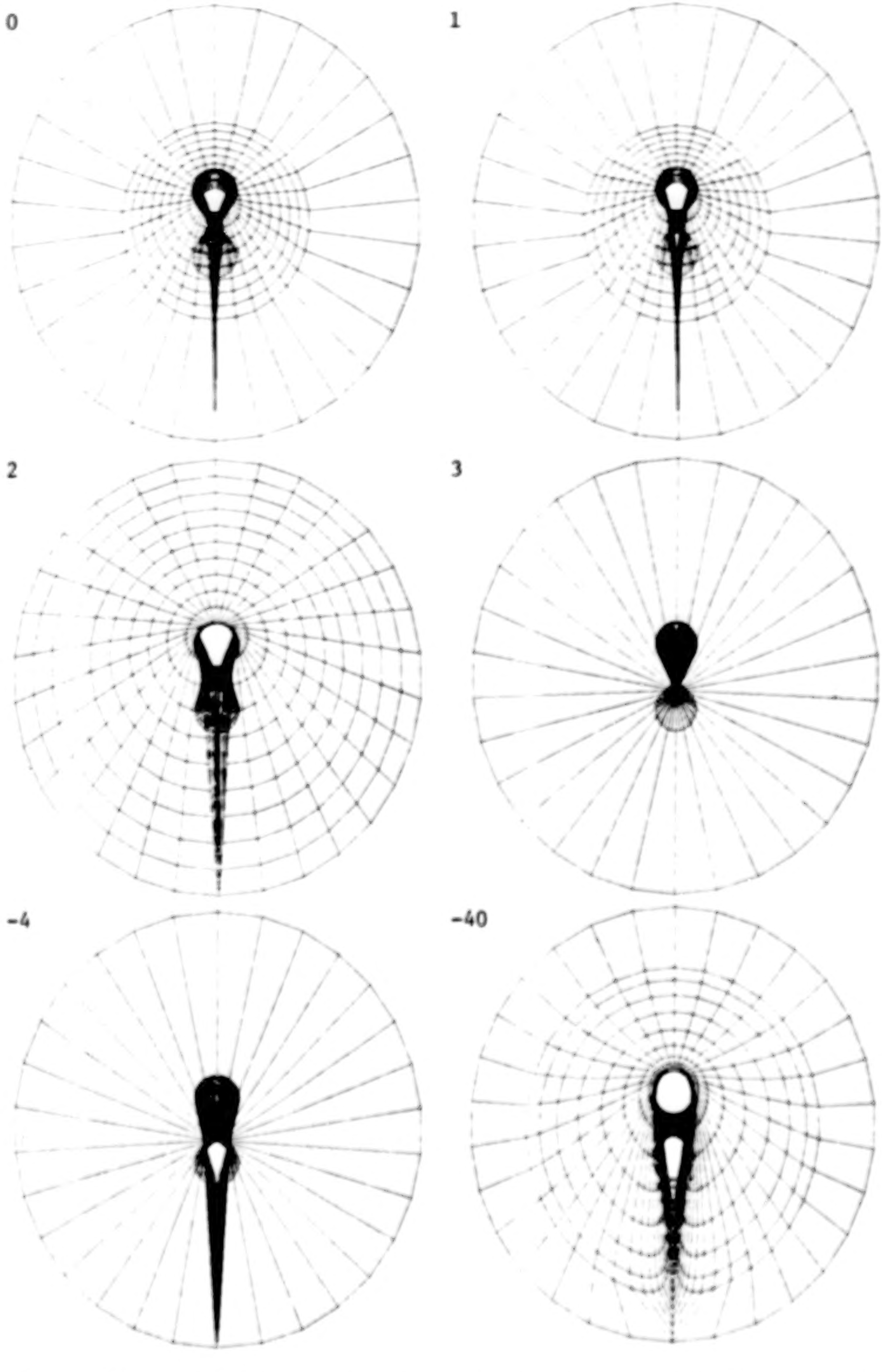


Figure 21. Initial Guess Types - Double-Body Segment Configuration #7

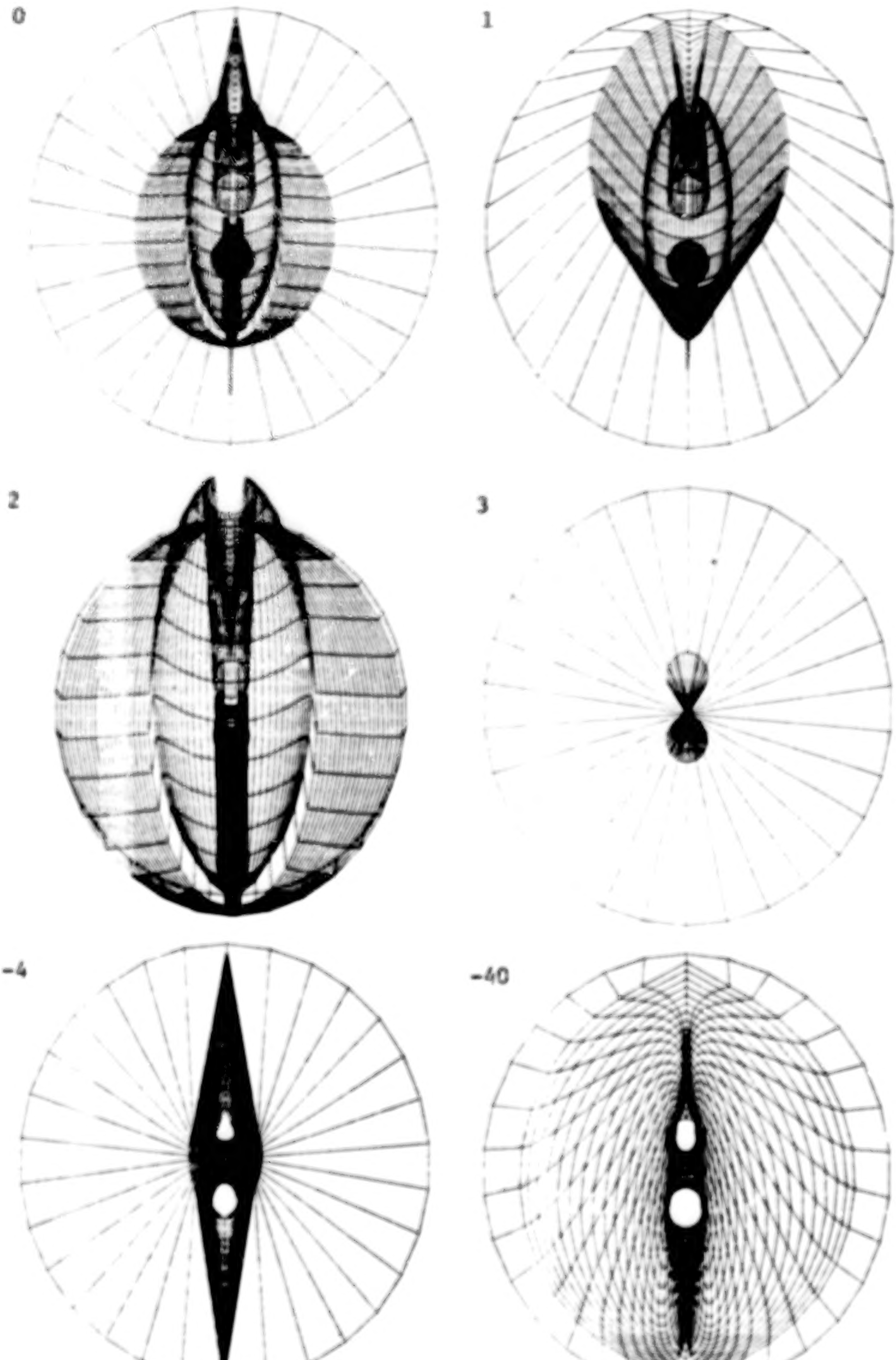


Figure 22. Initial Guess Types - Double-Body Segment Configuration #8

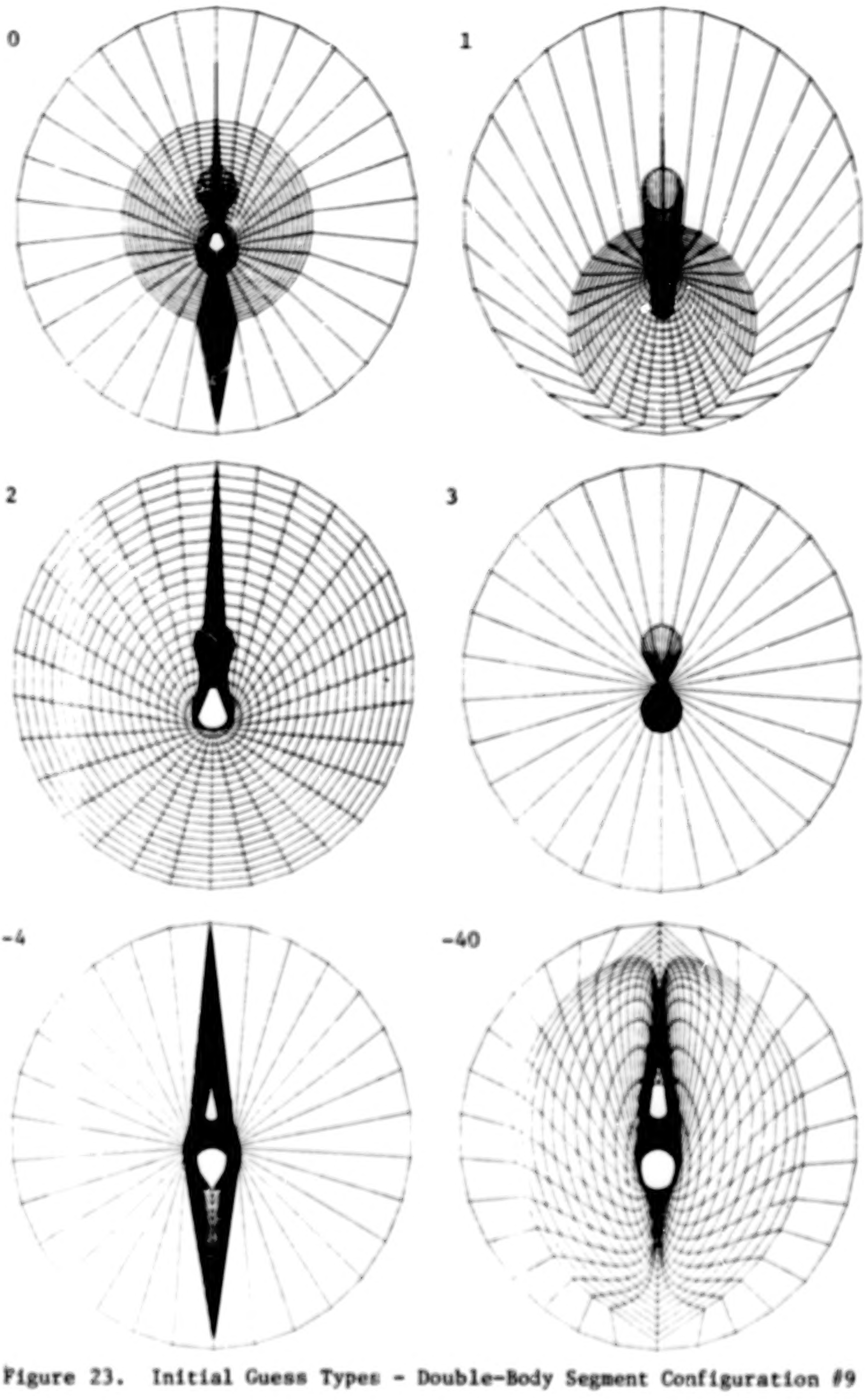


Figure 23. Initial Guess Types - Double-Body Segment Configuration #9

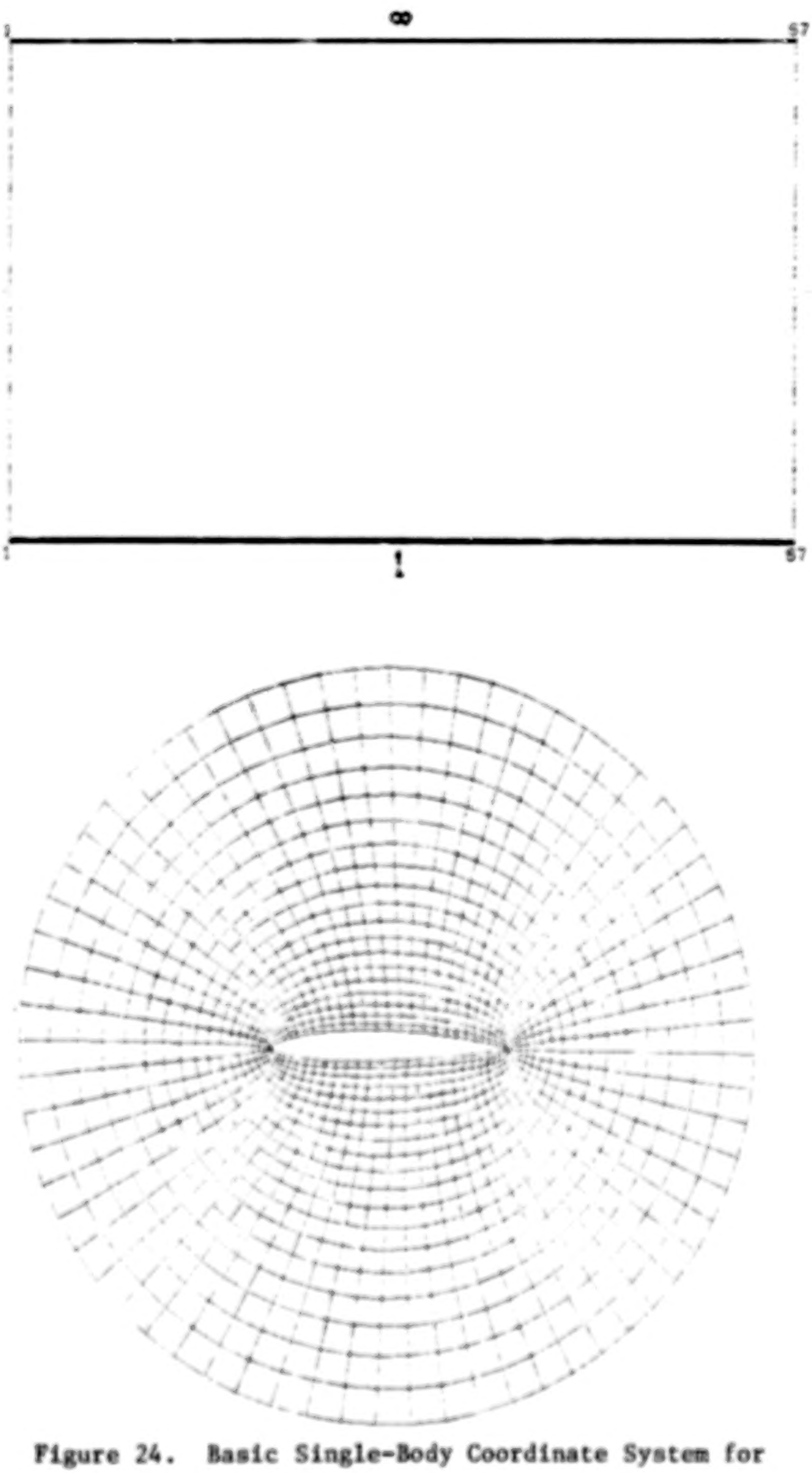


Figure 24. Basic Single-Body Coordinate System for Acceleration Parameter Studies

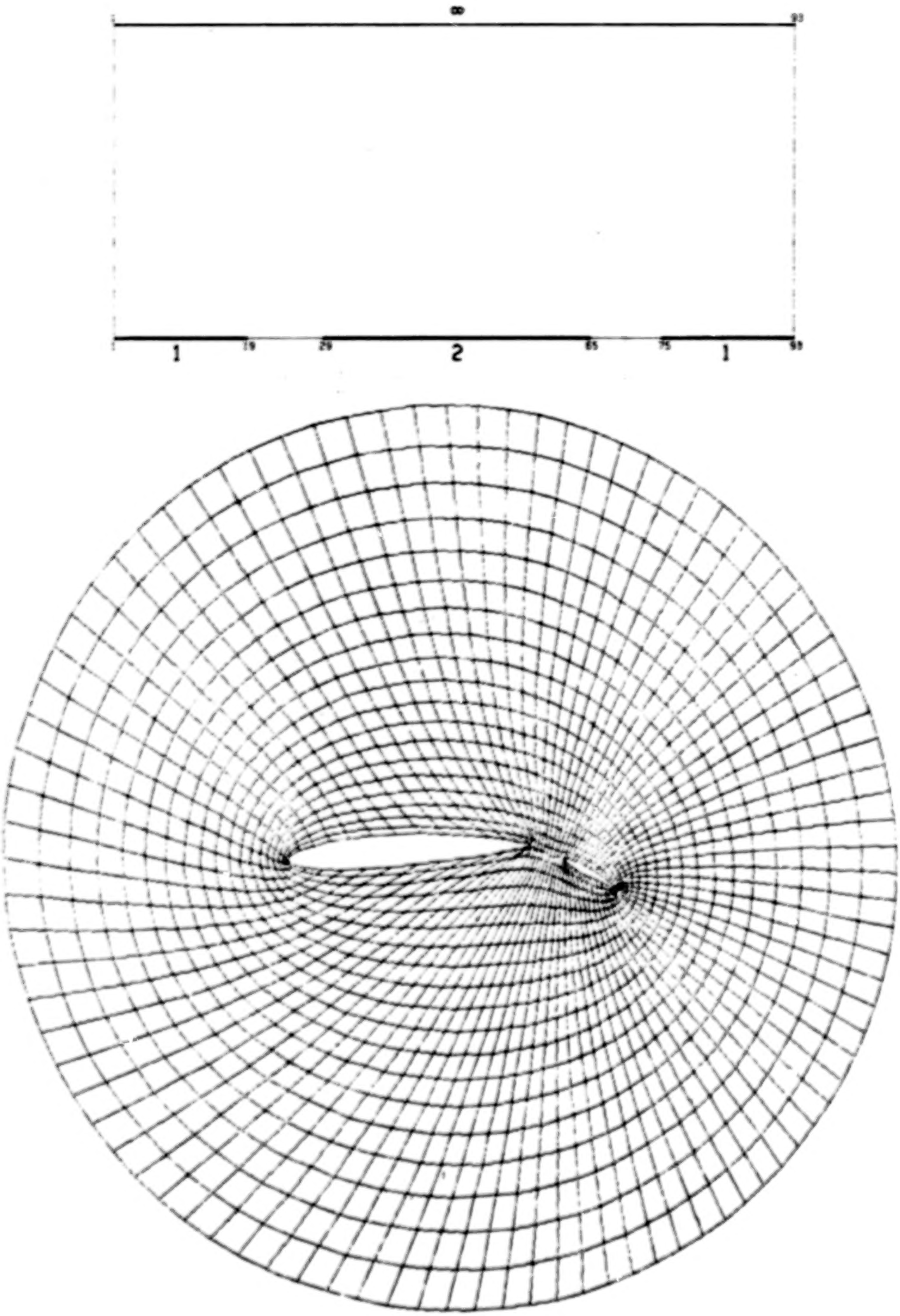


Figure 25. Basic Double-Body Coordinate System for Acceleration Parameter Studies

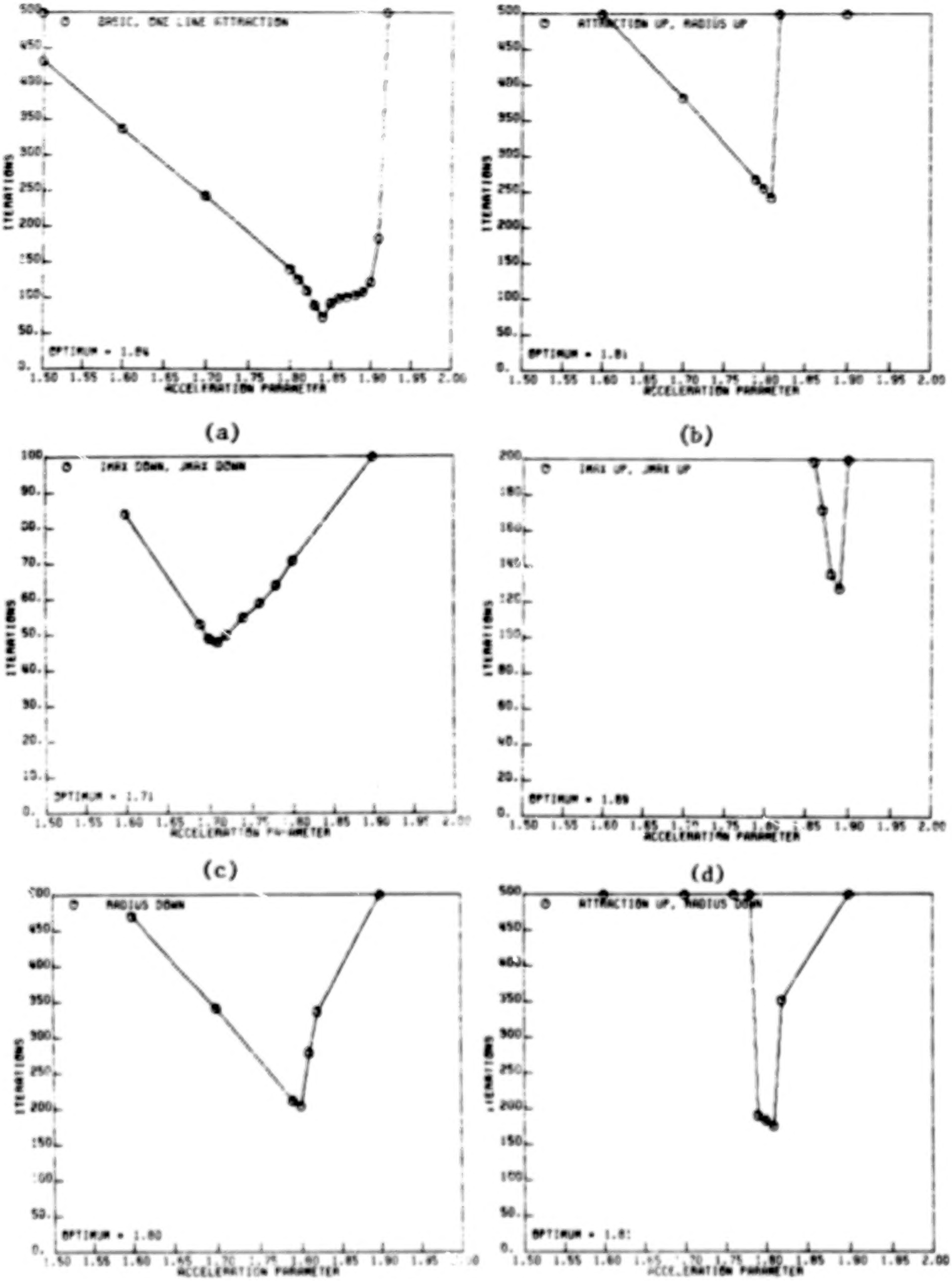


Figure 26. Effect of Acceleration Parameter on Convergence Rate

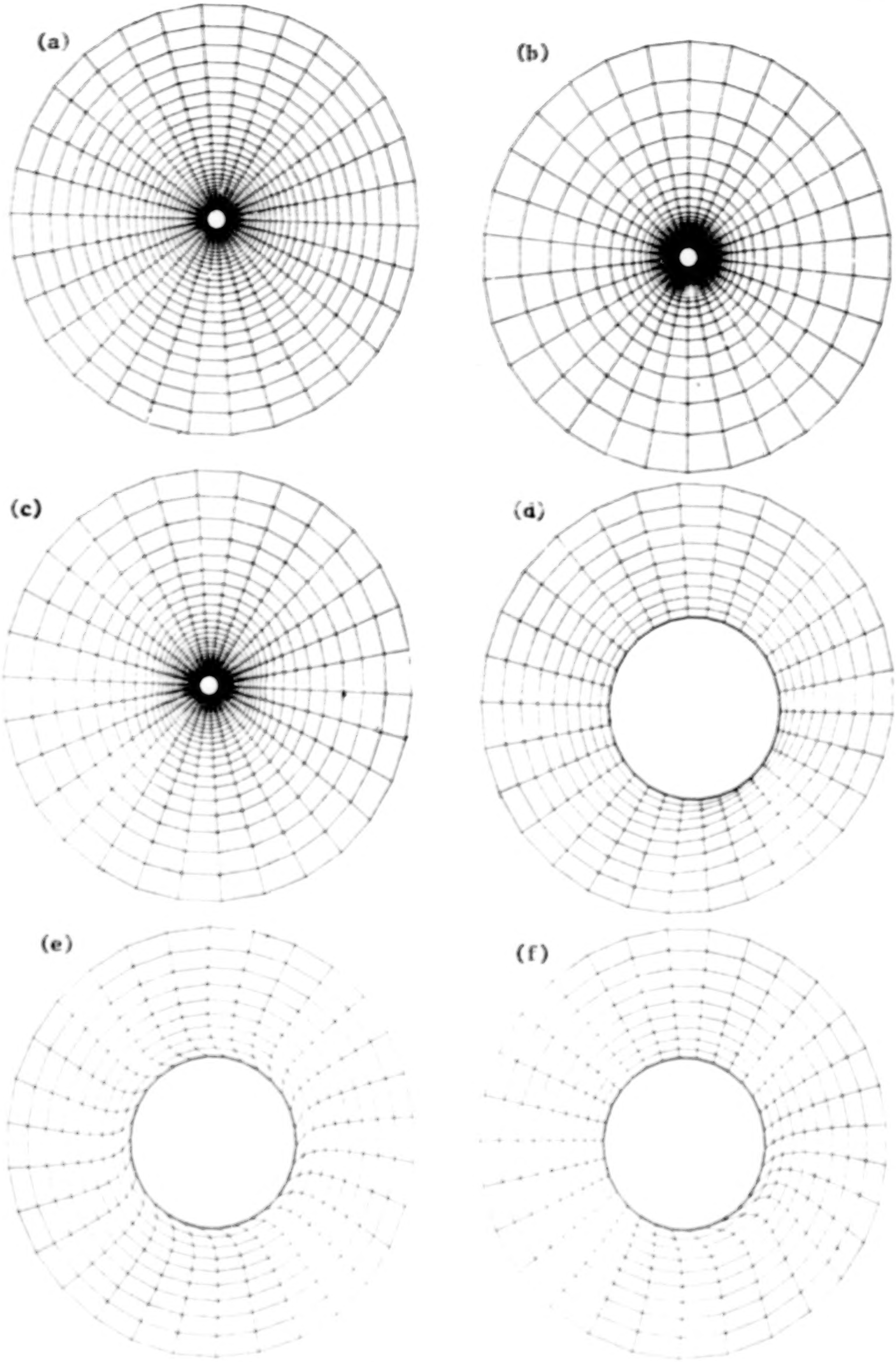


Figure 27. Examples of Coordinate System Control

(See Table 11 for control parameters.)

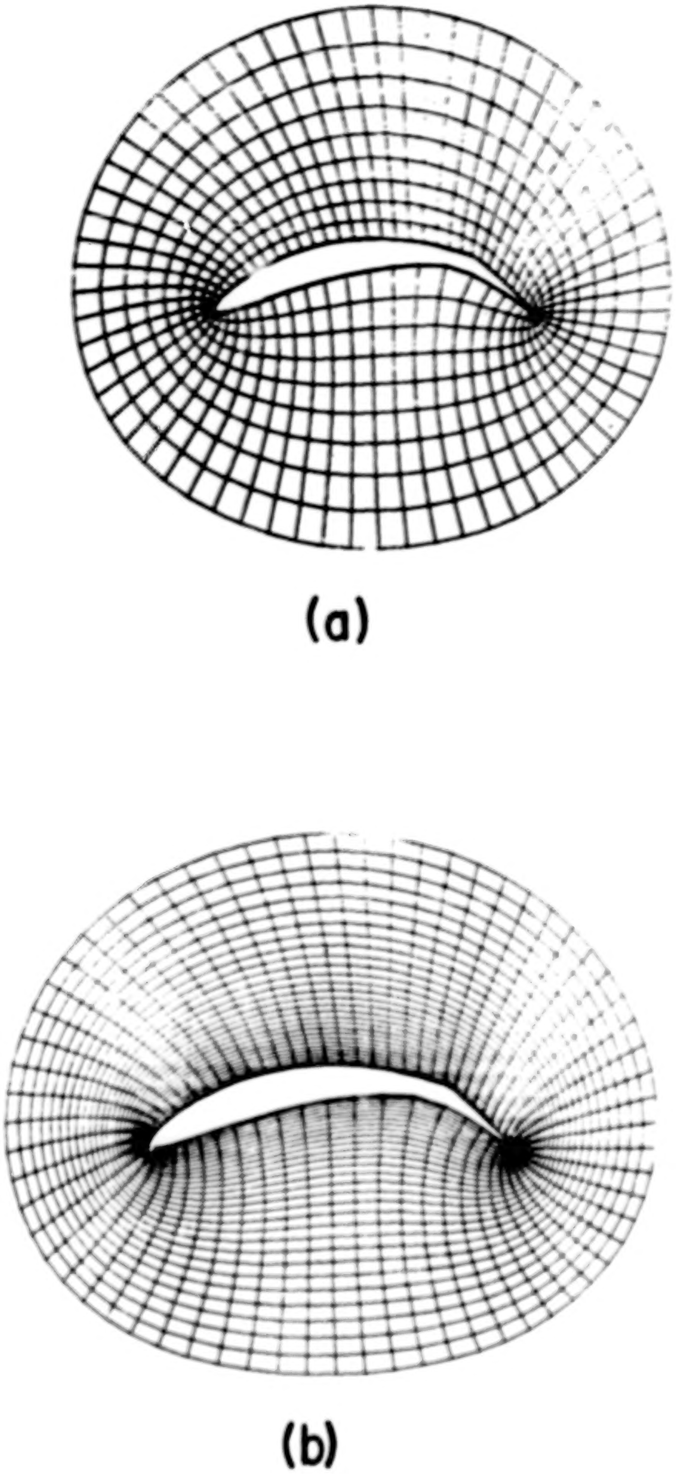


Figure 28. Example of Attraction into Concave Region

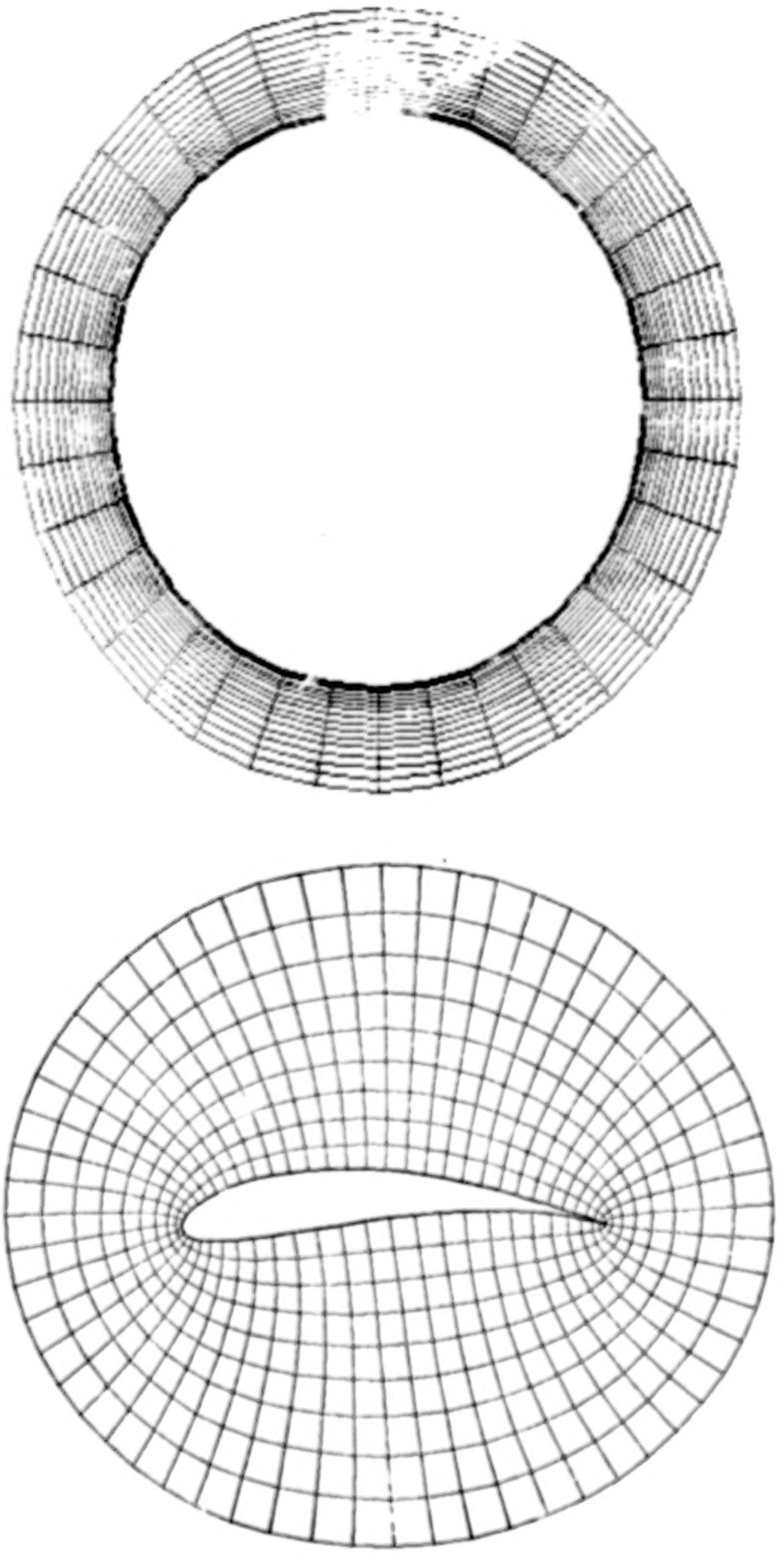
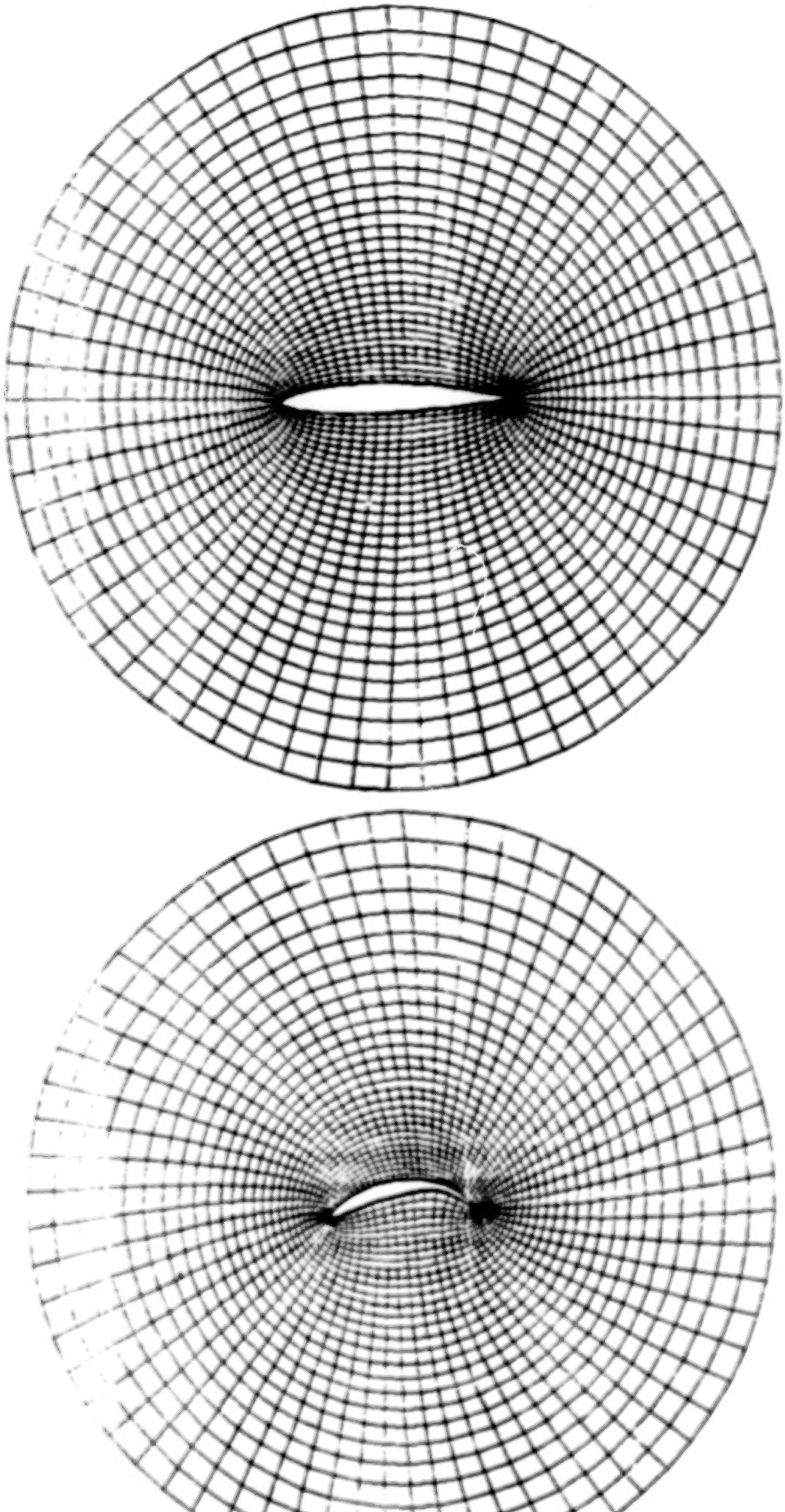


Figure 29. Coordinate Systems - Circular Cylinder And Joukowski Airfoil

(a)



(b)

Figure 30. Coordinate Systems - Karman-Trefftz Airfoils, No Coordinate Line Contraction

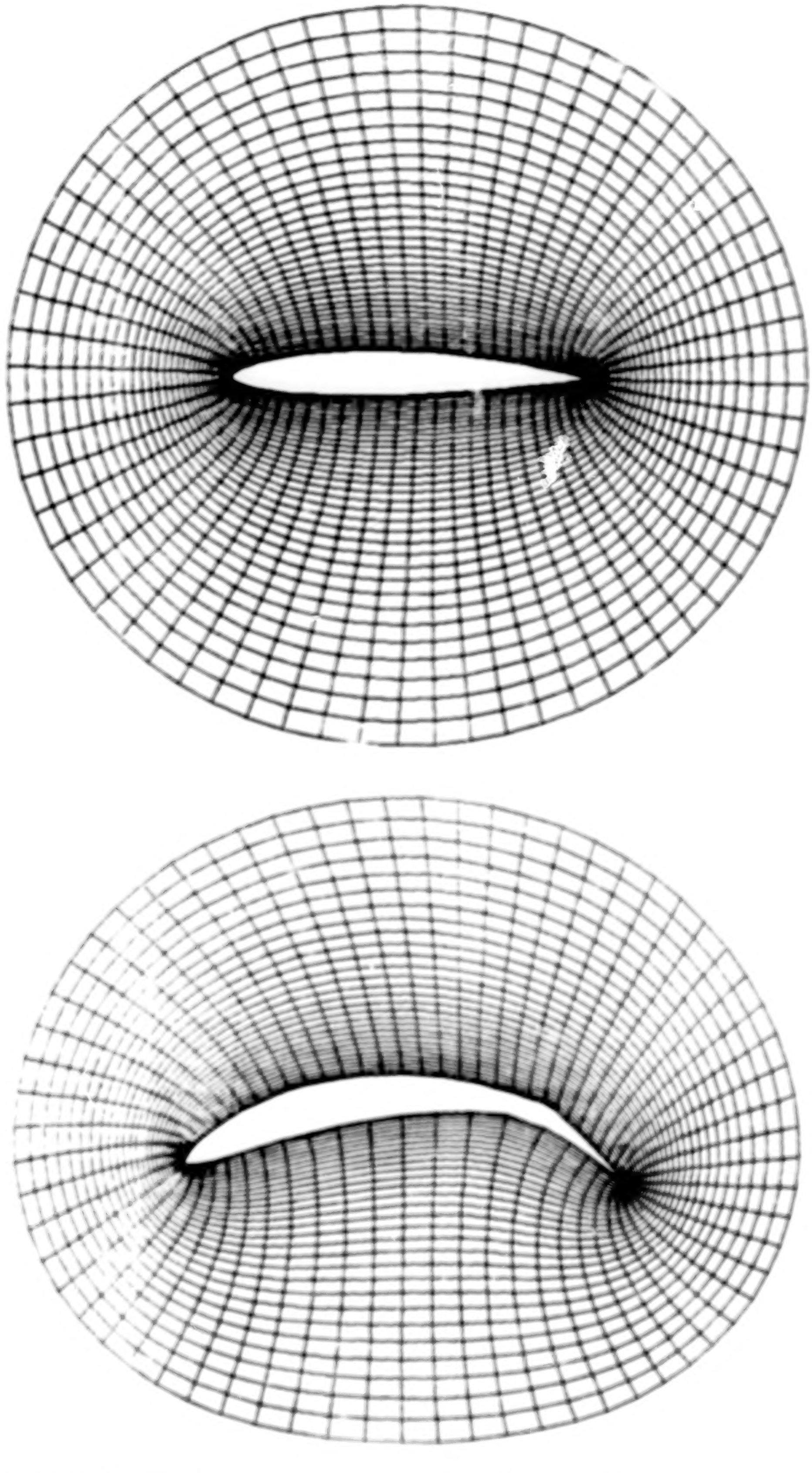


Figure 31. Coordinate Systems - Karman-Trefftz Airfoils, With Coordinate Line Contraction

202

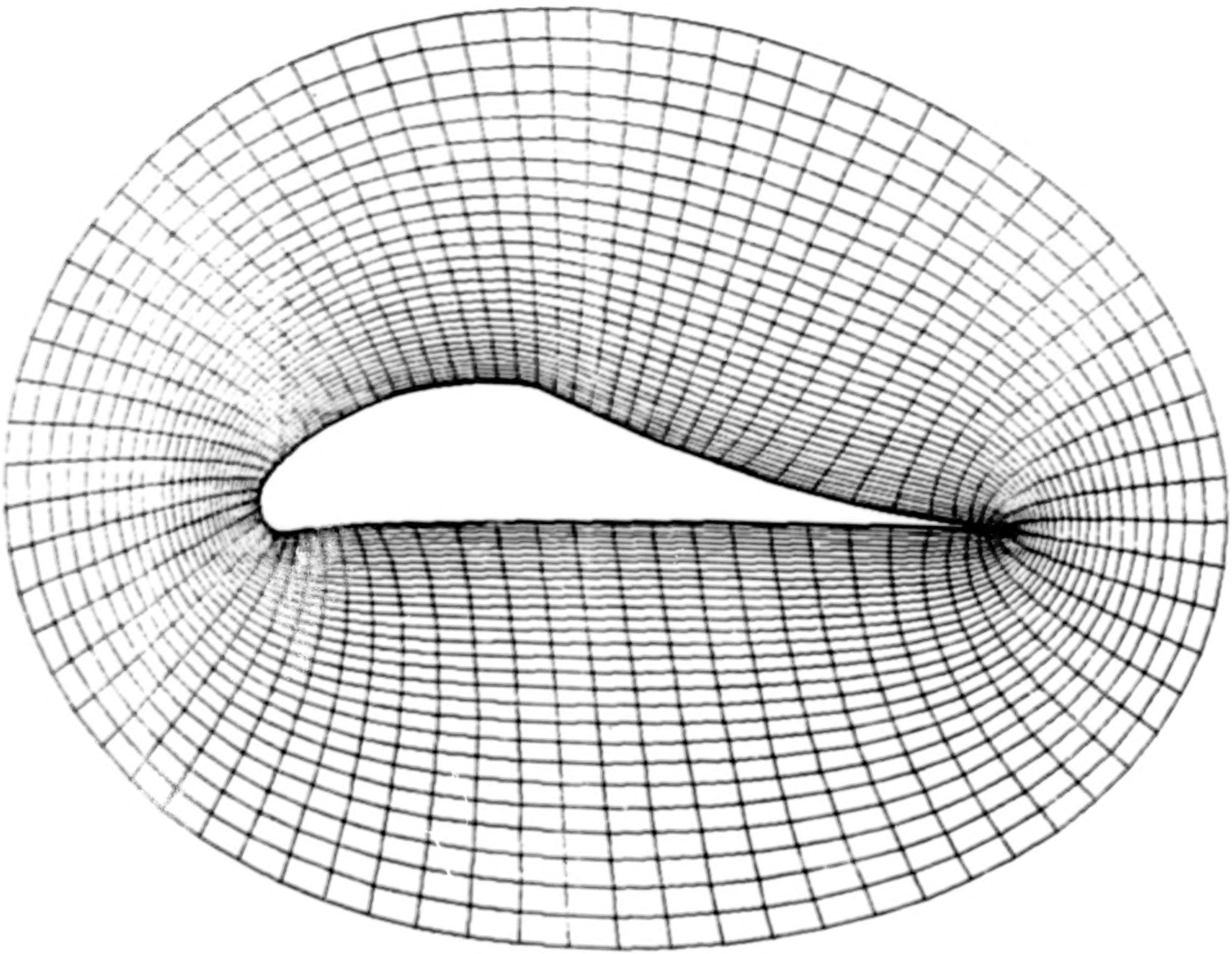


Figure 32. Contracted Coordinate System - Liebeck Airfoil

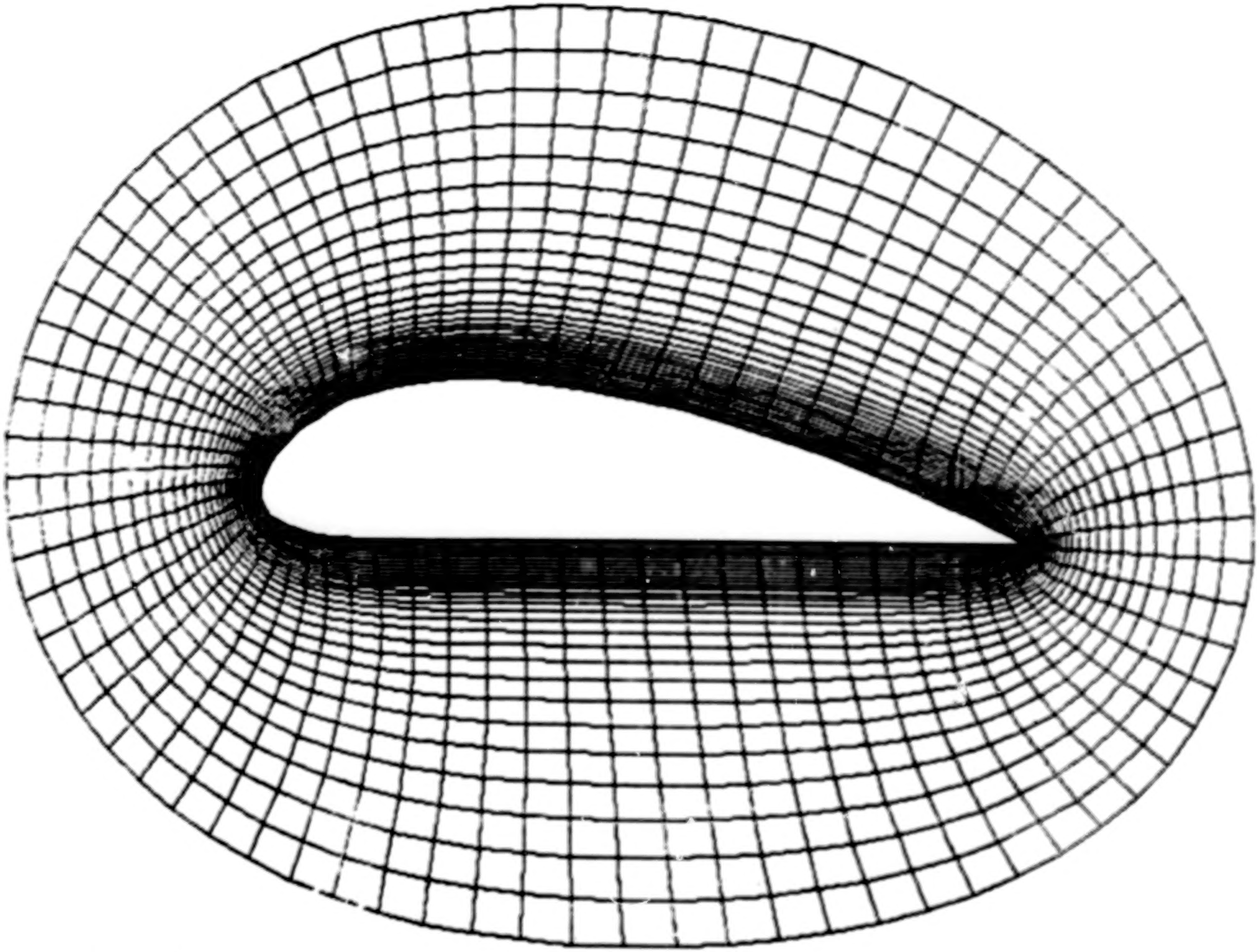


Figure 33. Contracted Coordinate System - Göttingen 625 Airfoil (See Table 12.)

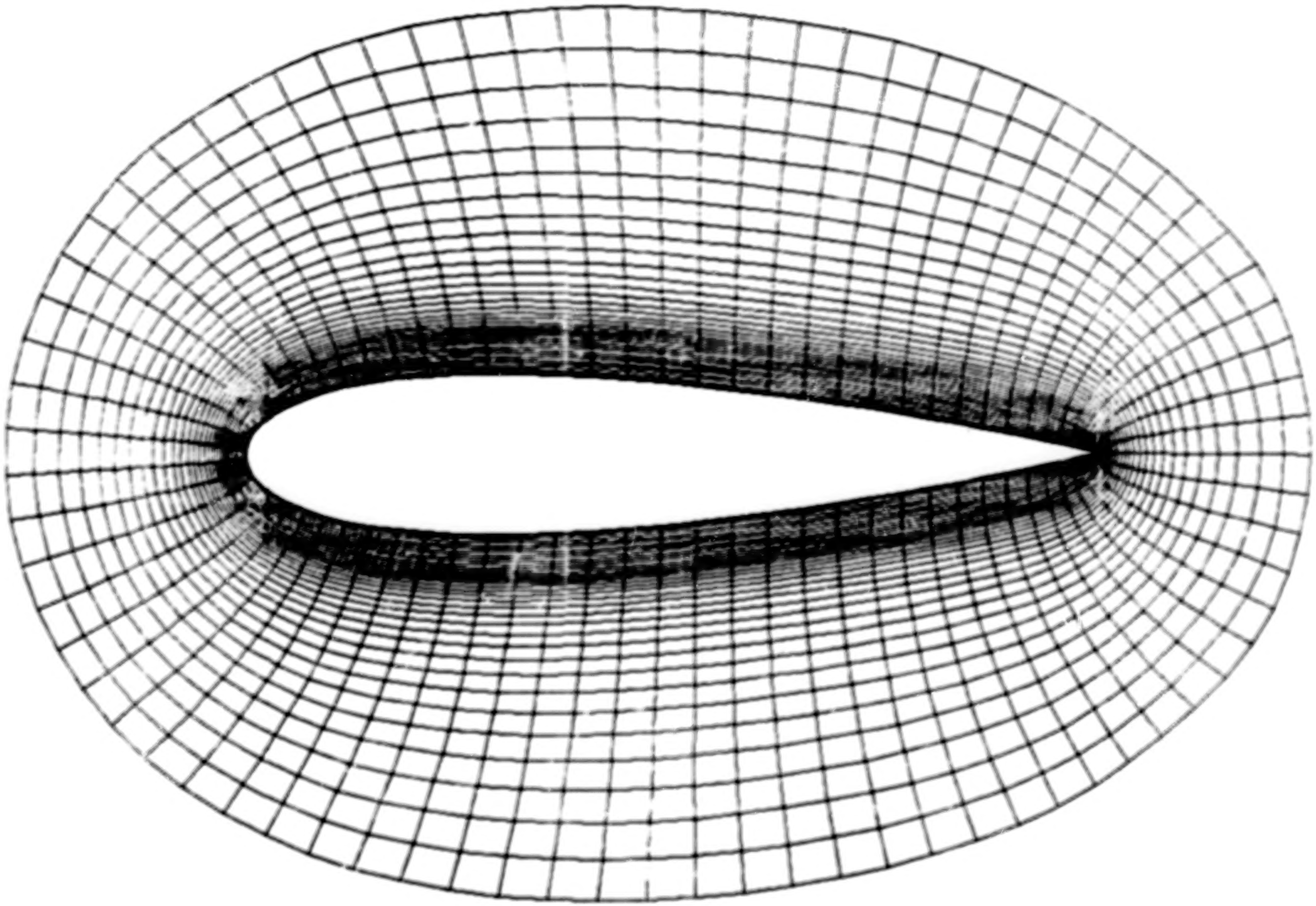


Figure 34. Contracted Coordinate System - NACA 0018 Airfoil (See Table 12.)

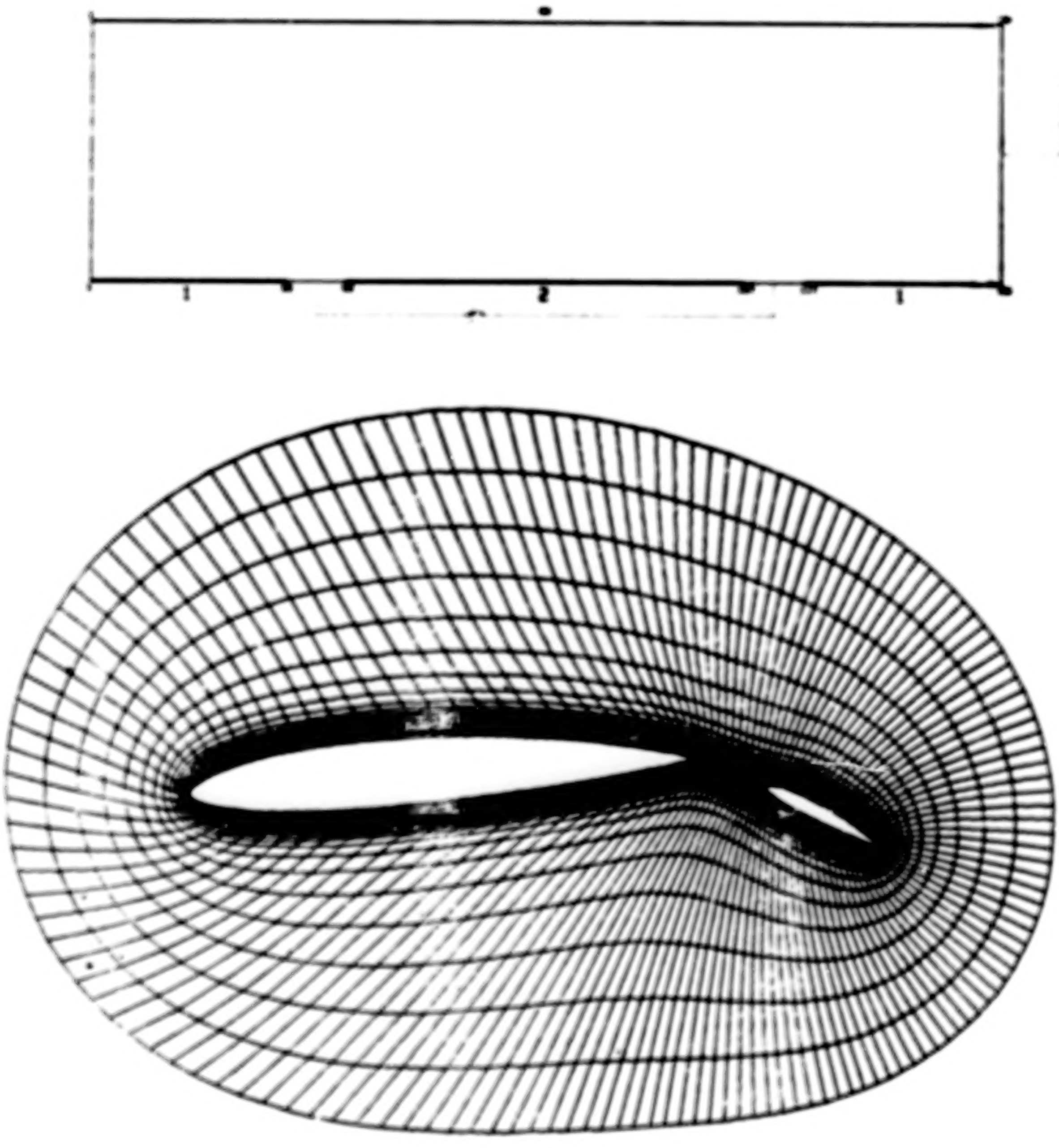


Figure 35. Contracted Coordinate System - Multiple Airfoil

Coordinate line attraction to the first 10 lines around the airfoils with amplitude 10,000 on the body. Decay factor of 1.0 for all except last line, where 0.5 was used. 37 points on airfoils, 40 lines around airfoil. Circular boundary of radius 10 fore body chords.

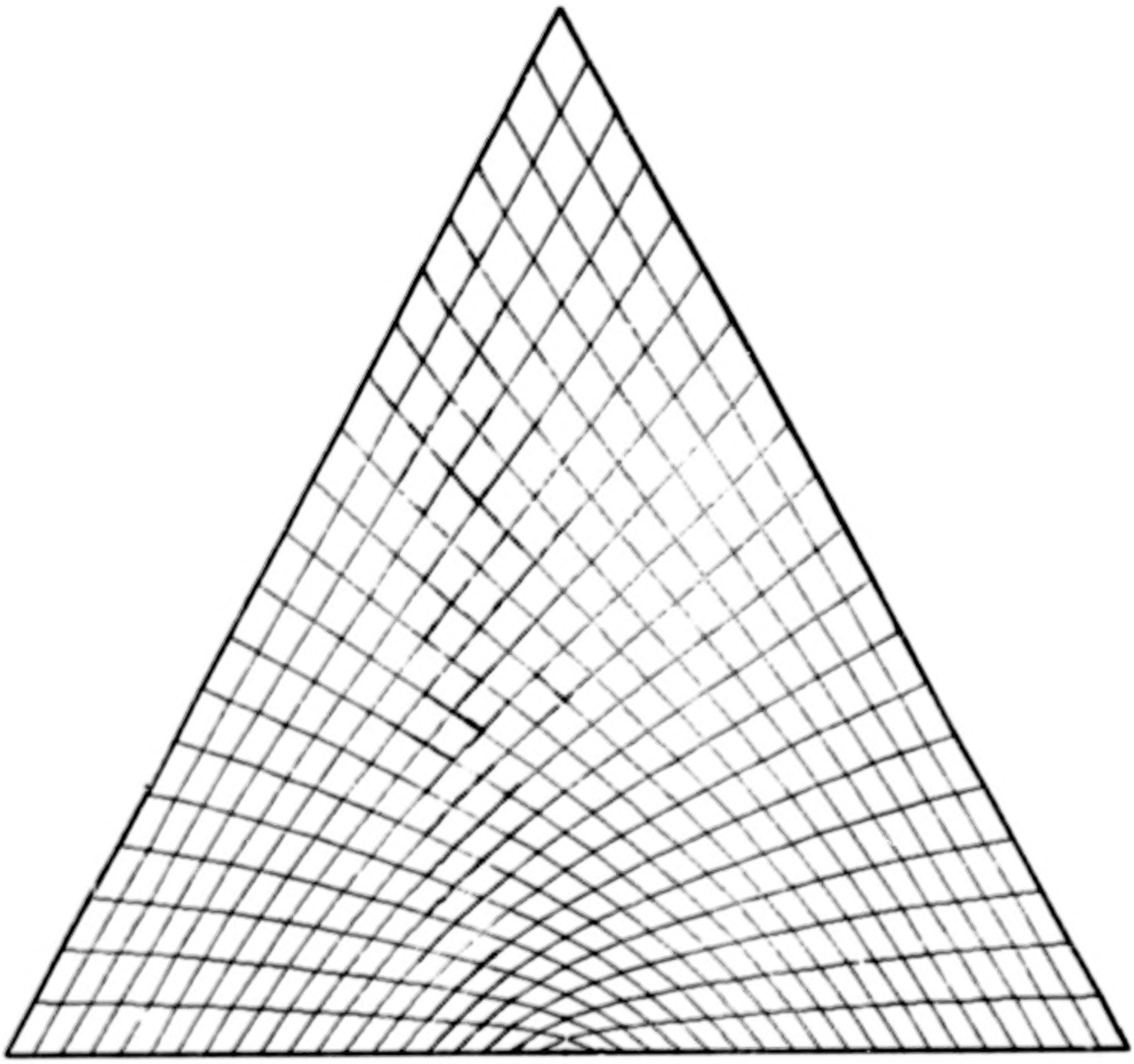


Figure 36. Coordinate System - Triangular Simply-Connected Region.

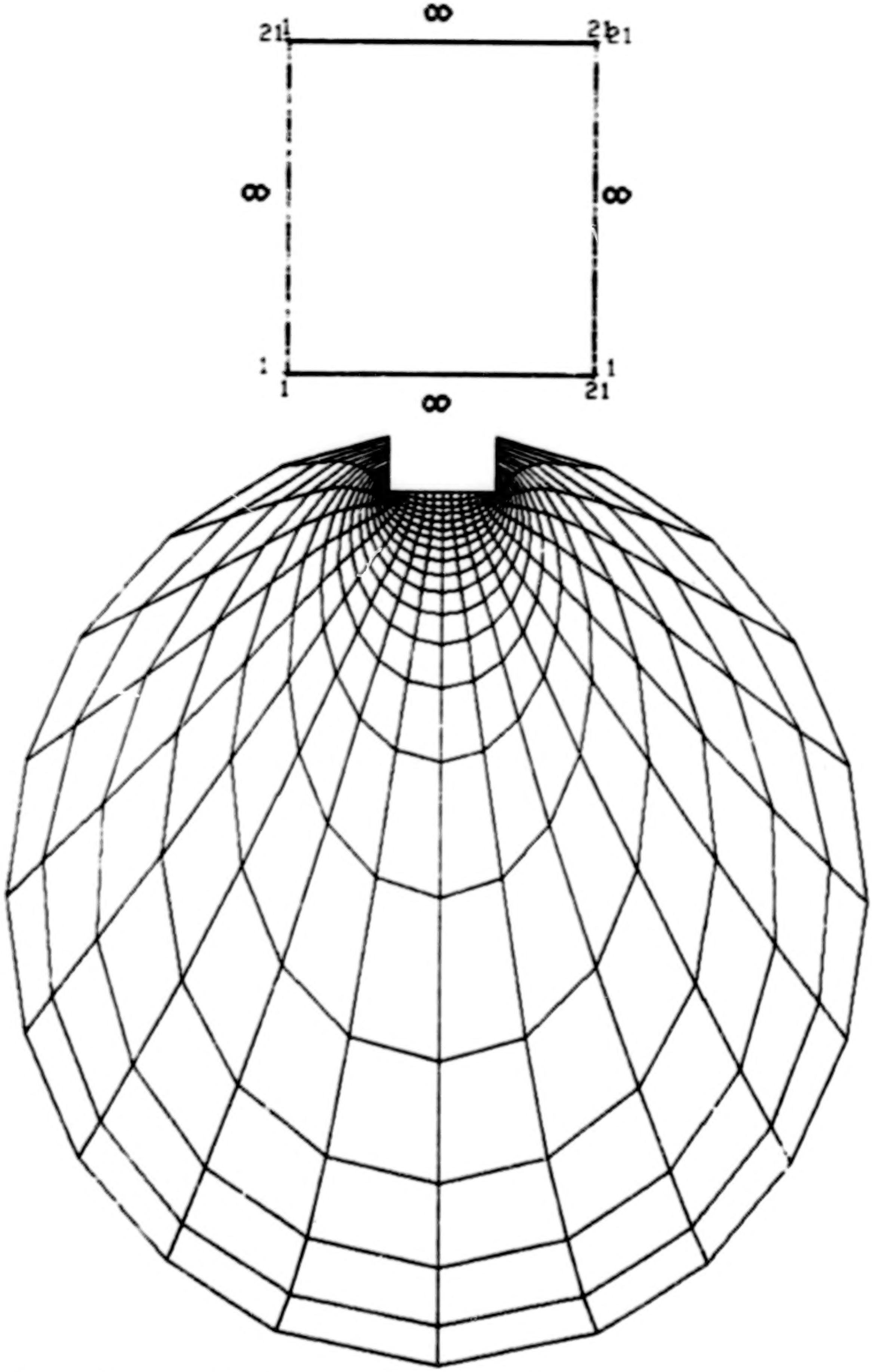


Figure 37. Coordinate System - Key-seat Shaft, Simply-Connected Region (See Table 12)

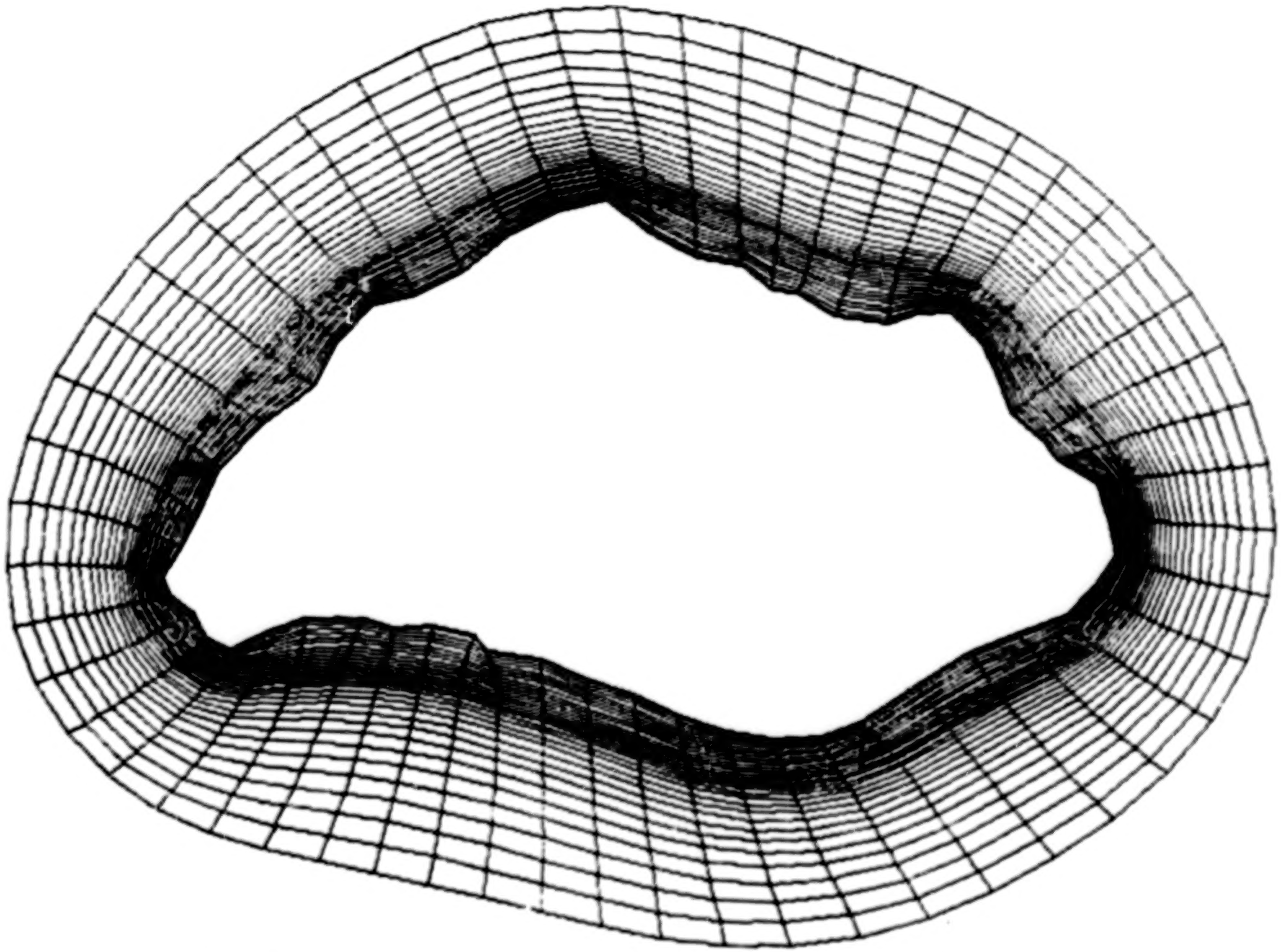
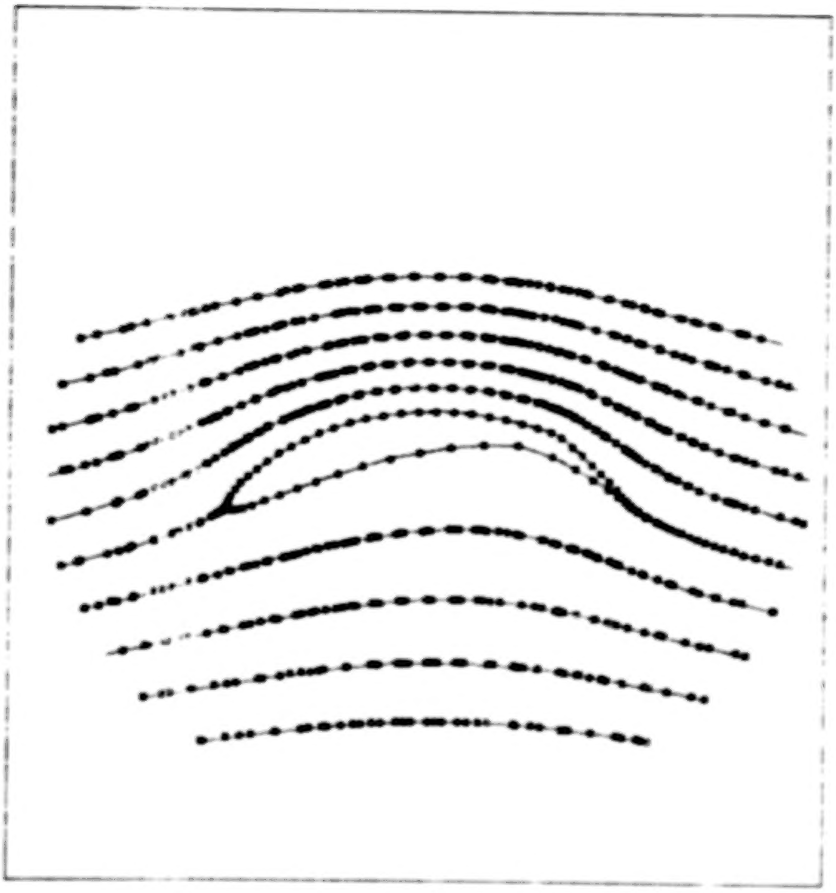


Figure 38. Contracted Coordinate System - Rock (See Table 12.)



LIFT COEFFICIENT = 2.595696

DRAG COEFFICIENT = 0.003966

LIFT COEFFICIENT ERROR = -0.005103

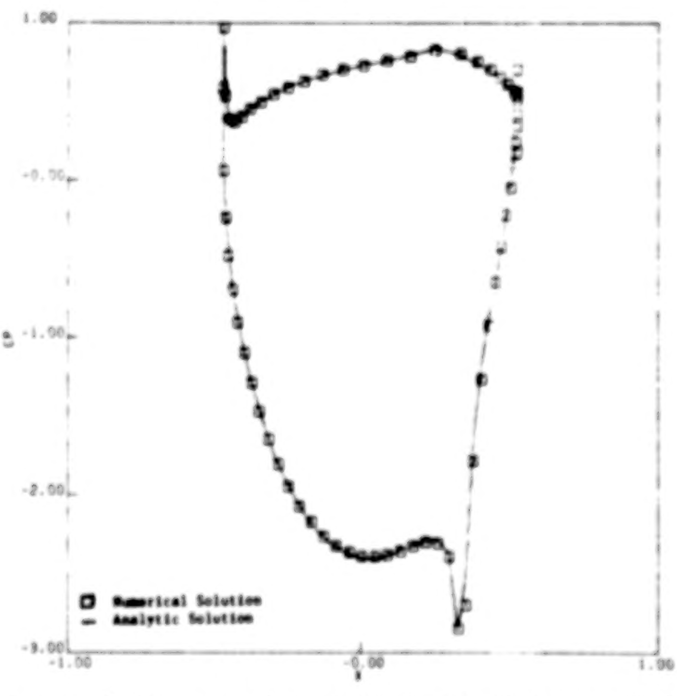


Figure 39. Comparison of Numerical and Analytic Potential Flow Solutions for Flapped Karman-Trefftz Airfoil. No coordinate attraction.

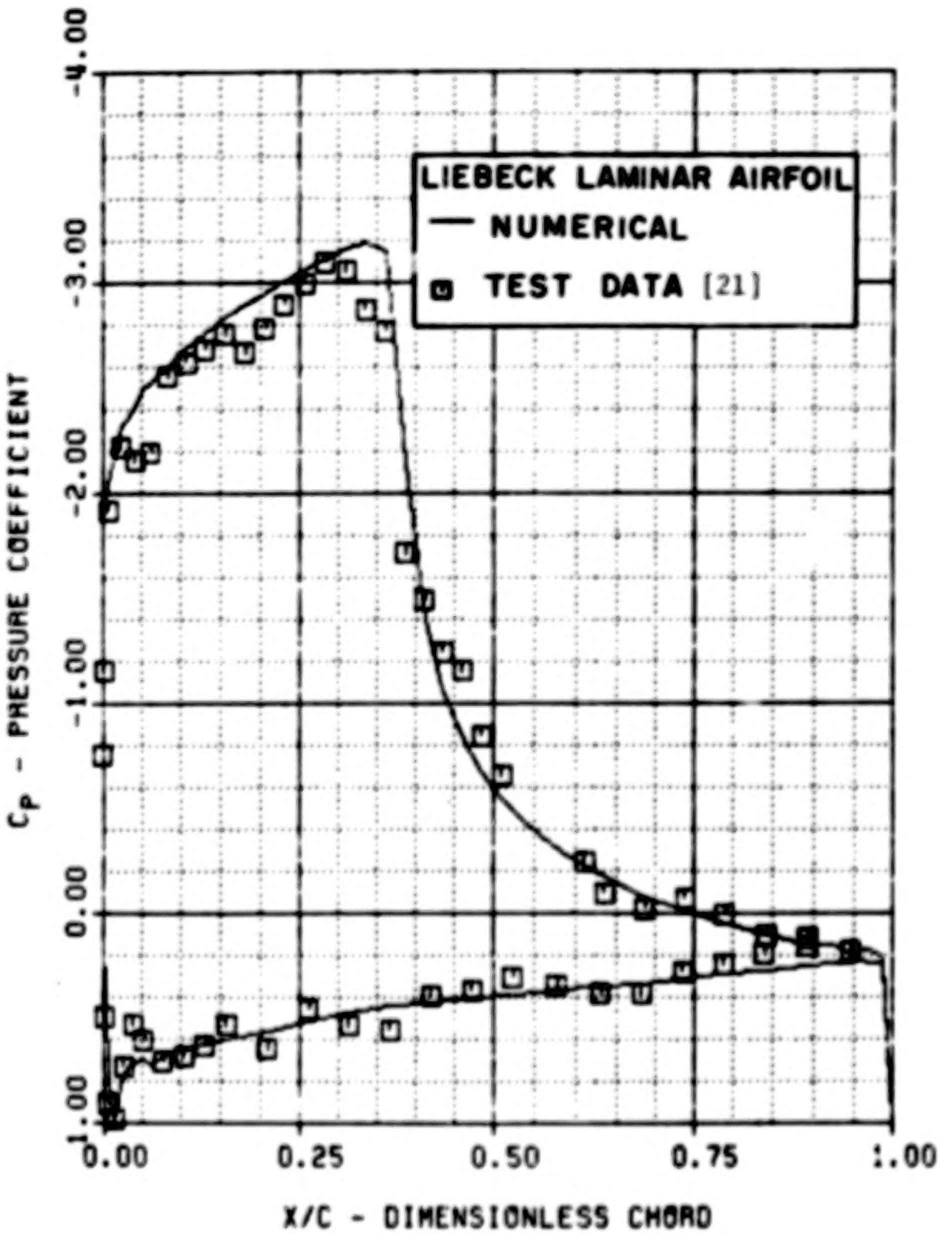


Figure 40. Comparison of Experimental and Numerical Pressure Distribution - Lieback Laminar Airfoil

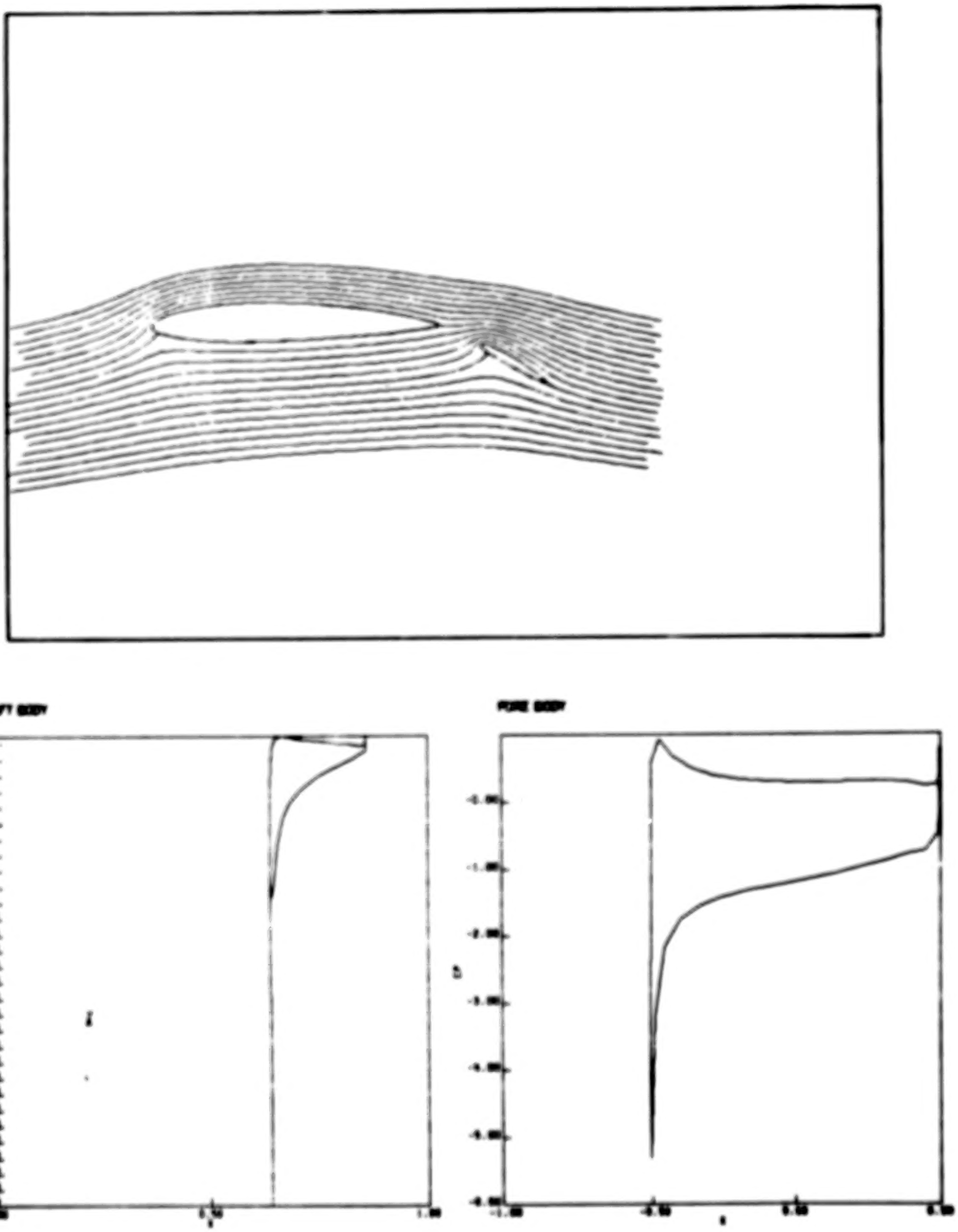


Figure 41. Potential Flow Streamlines and Pressure Distributions - Double Airfoil

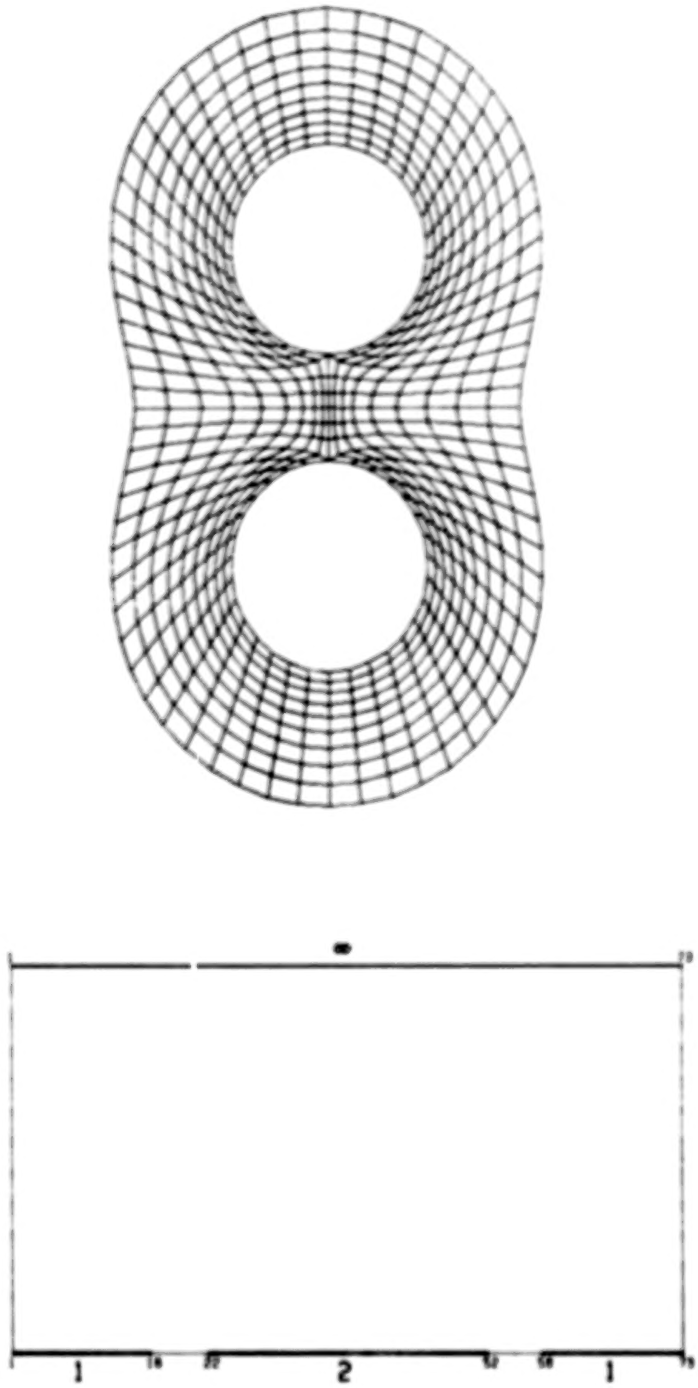
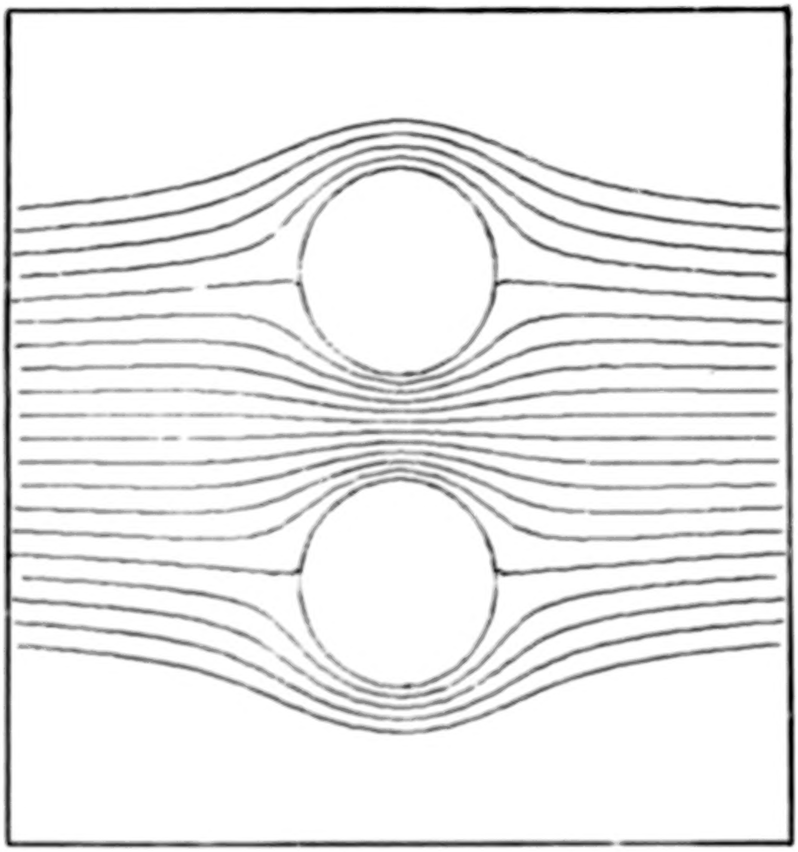


Figure 42. Coordinate System - Double Cylinders, Coordinate Attraction to Cut Intersections

Attraction to intersections of body contours with cut between at amplitude 1000 and decay factor 0.5. 31 points on each obdy. 40 lines around the bodies. Outer boundary at 20 diameters.

(a)



(b)

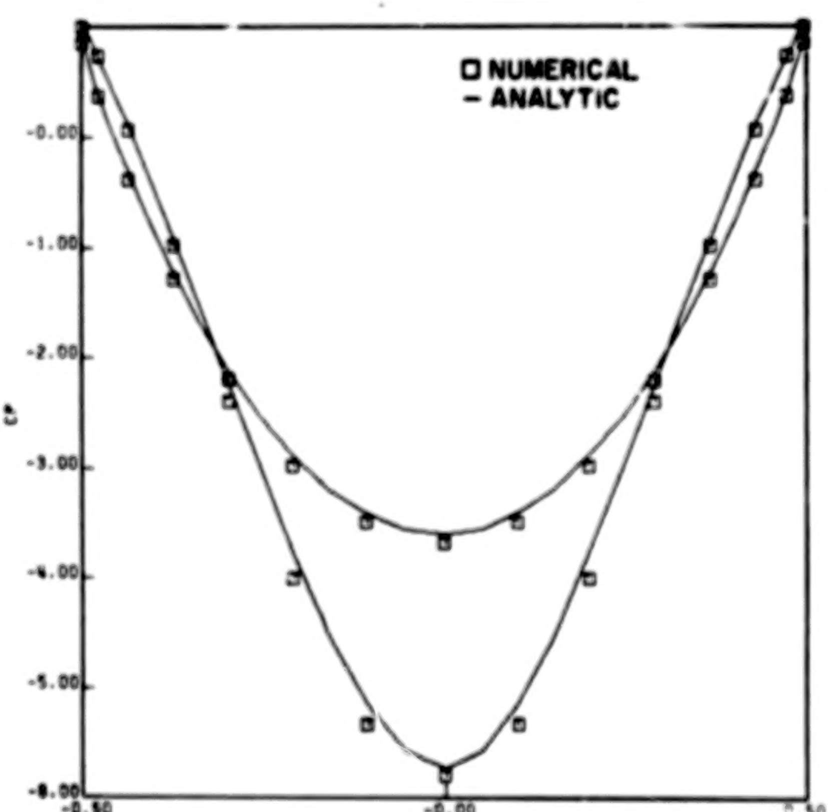


Figure 43. Potential Flow Streamlines and Pressure Distribution - Double Cylinder

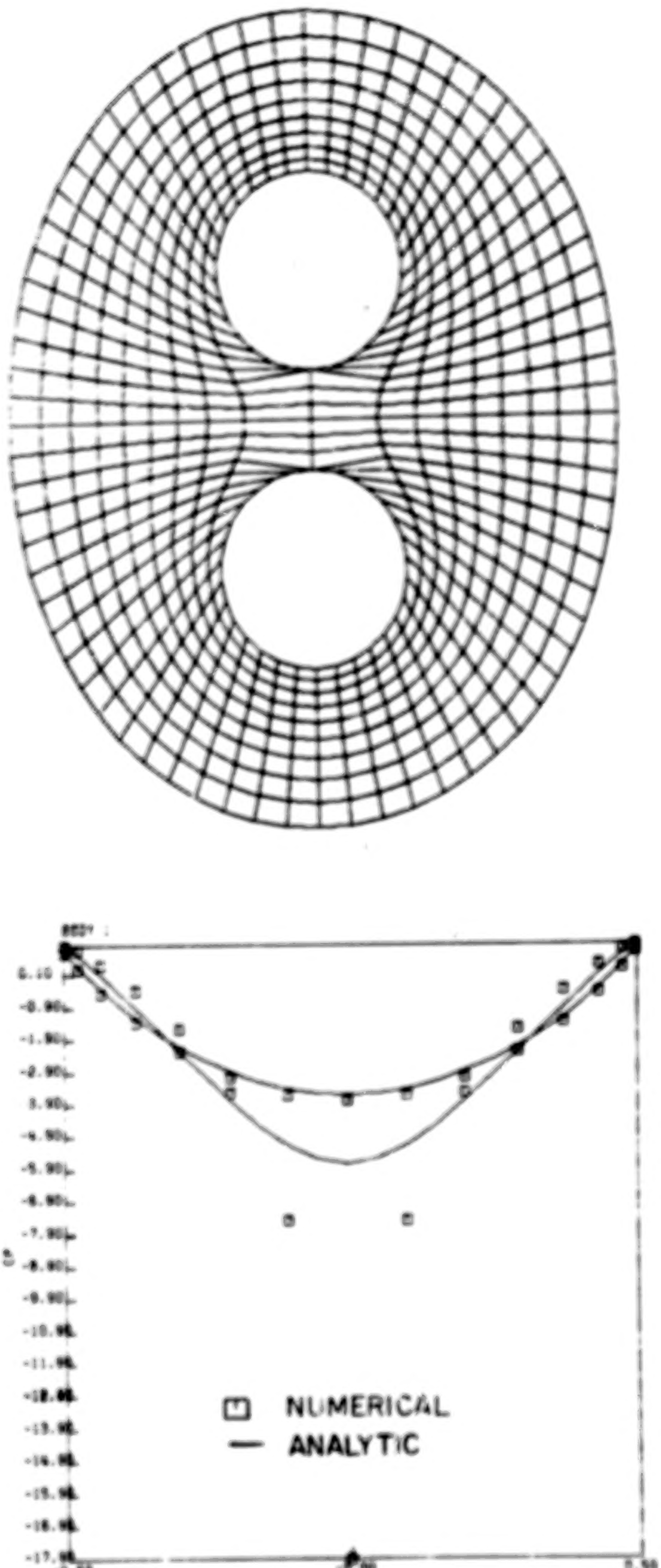


Figure 44. Potential Flow Pressure Distribution for Double Cylinder without Coordinate Attraction

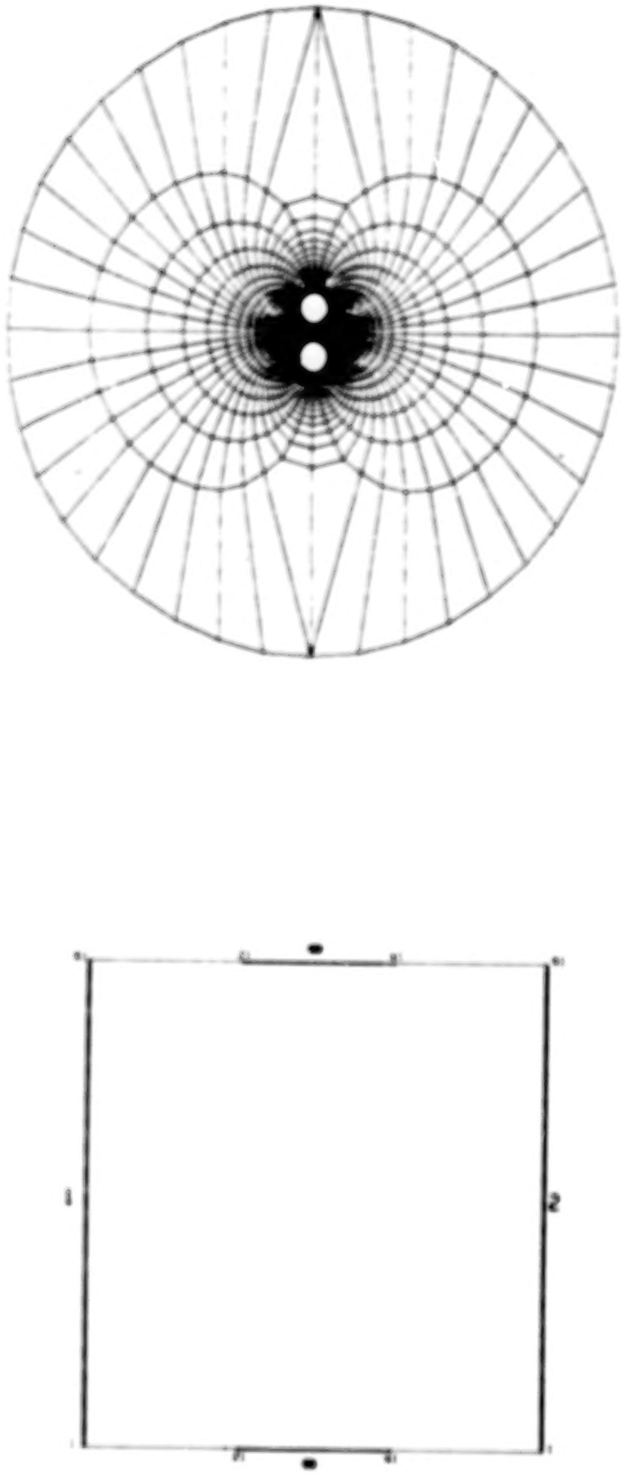


Figure 45. Alternate Segment Arrangement - Double Cylinders
No coordinate attraction.

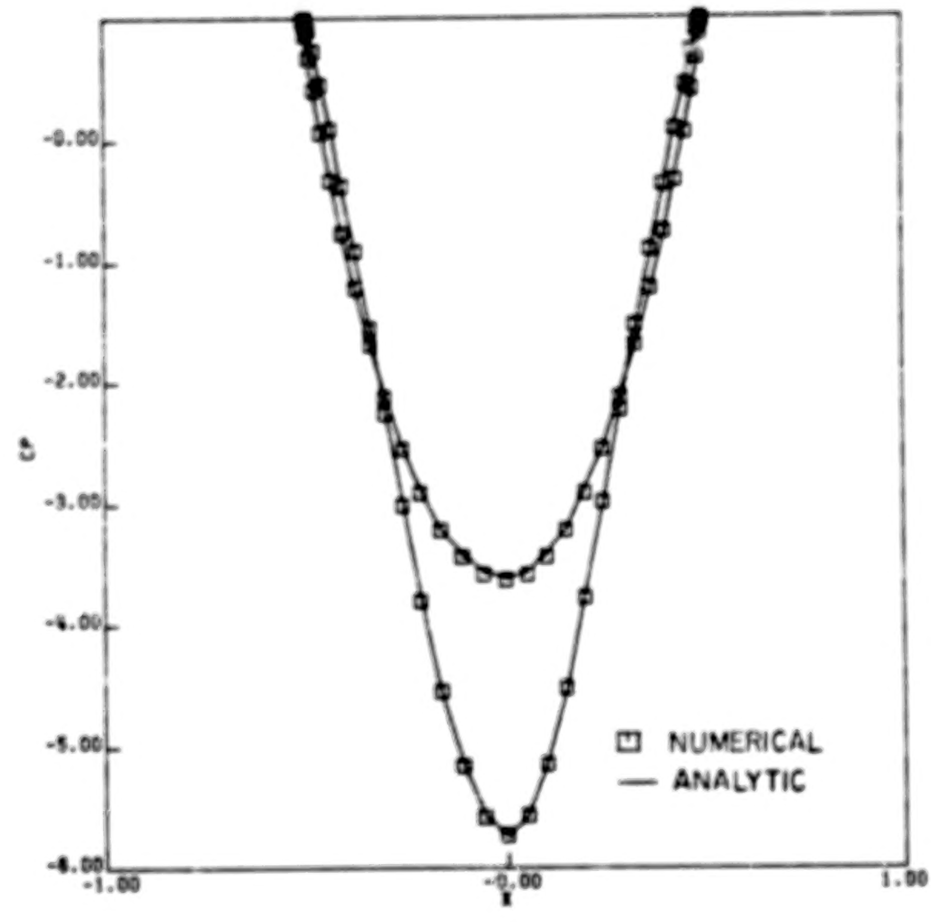
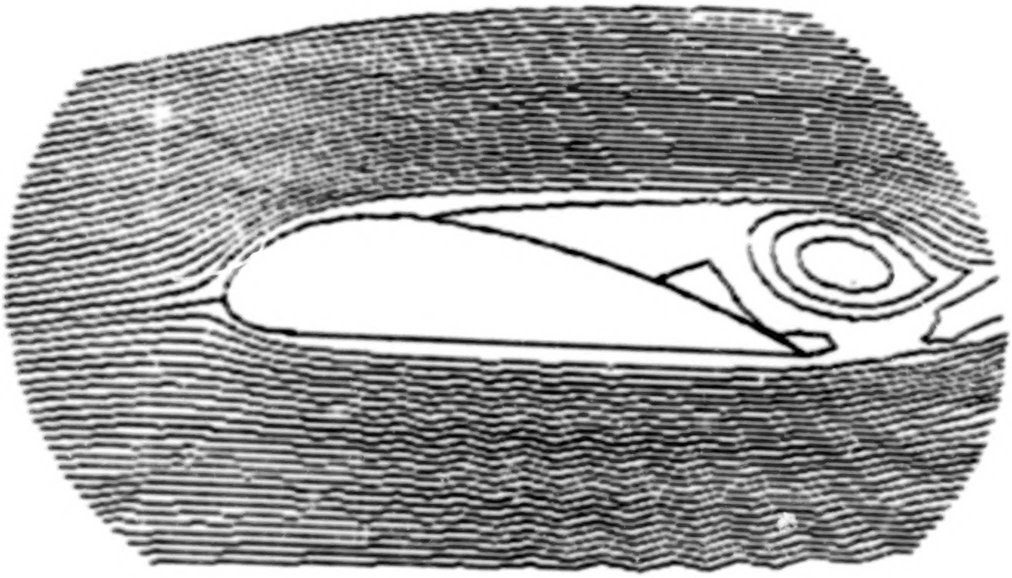
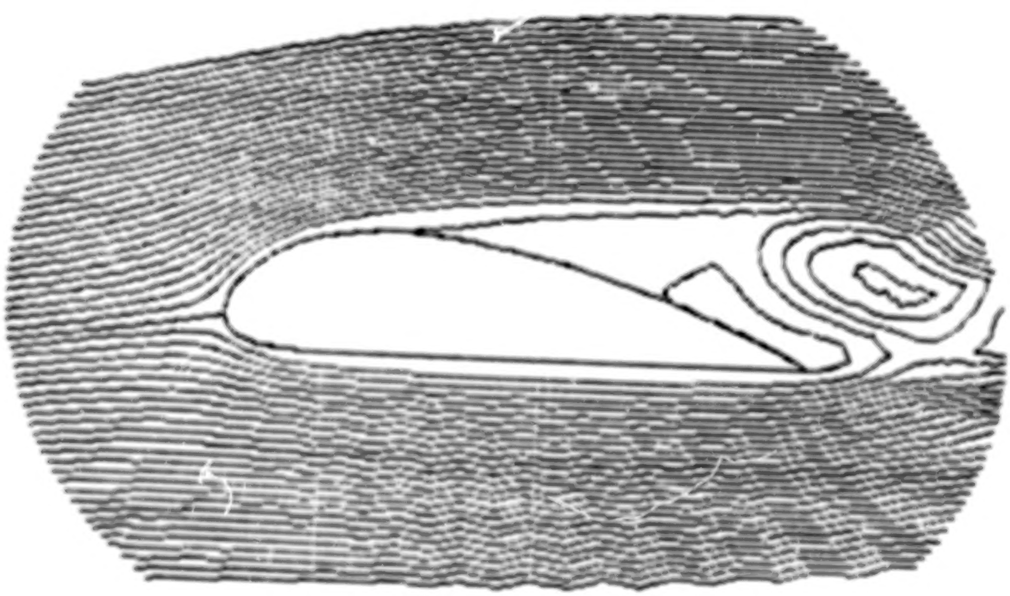


Figure 46. **Potential Flow Pressure Distribution for Double Cylinders - Alternate Segment Arrangement**



t=3.13



t=3.33

Figure 47. Streamlines - Göttingen 625 Airfoil, R = 2000, α = 5°

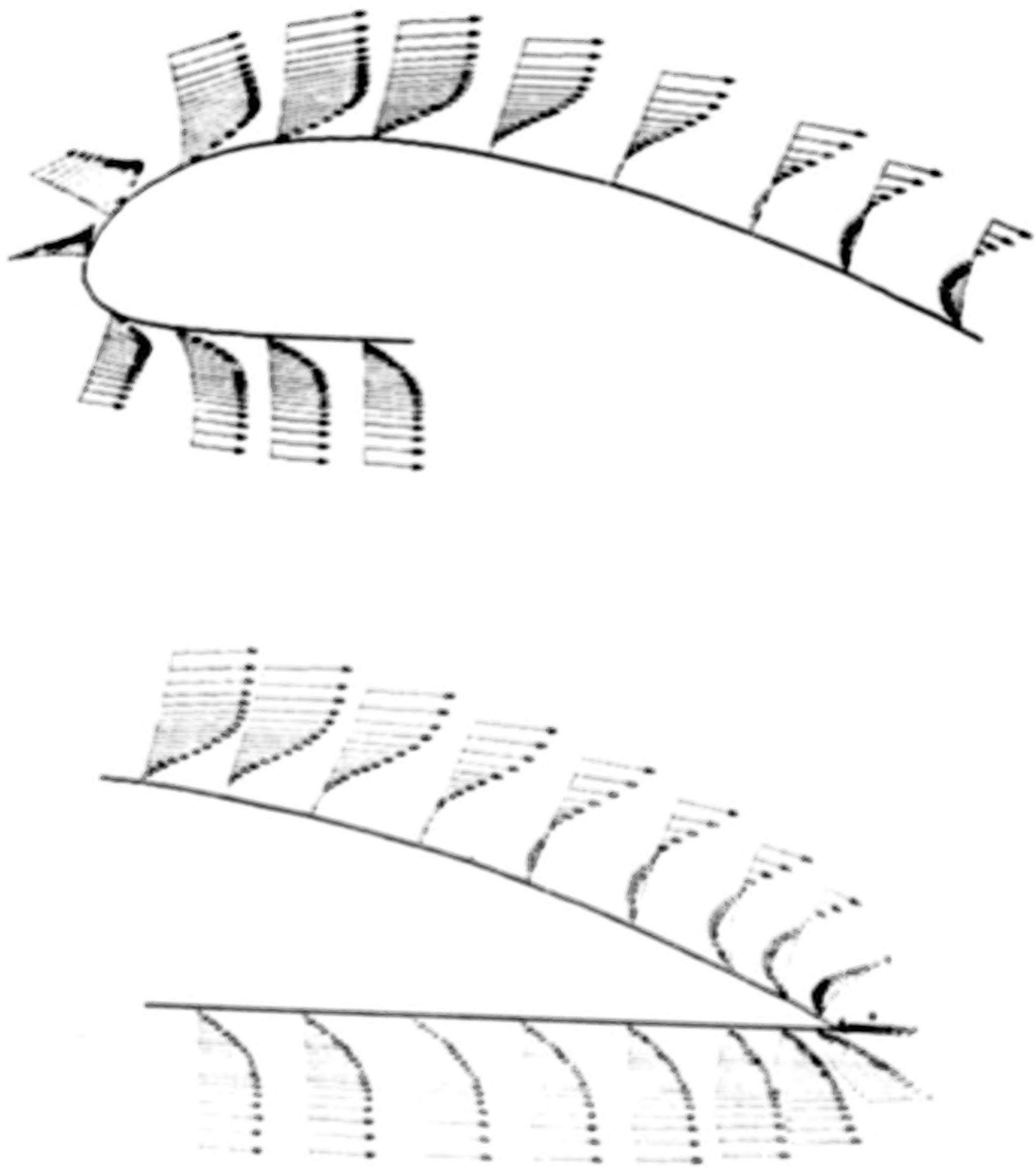
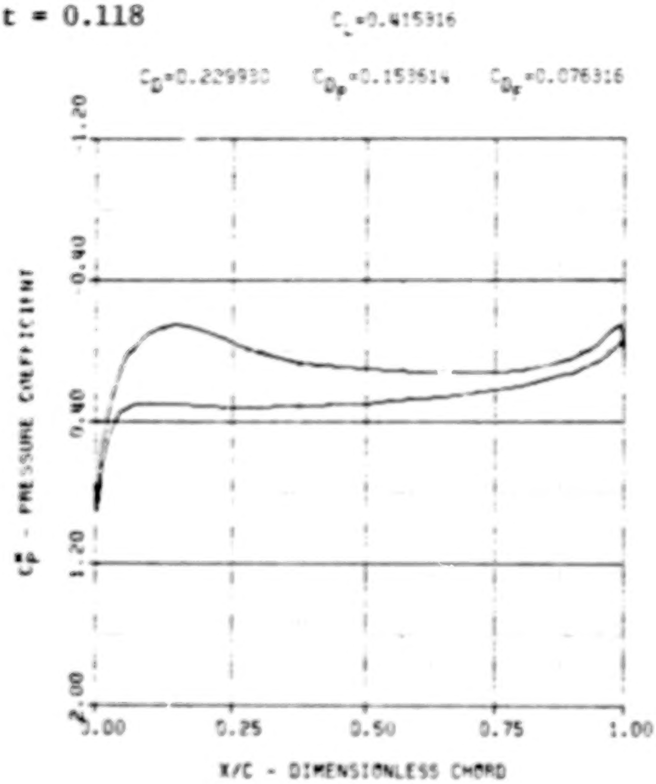


Figure 48. Velocity Profiles, Gottingen 625 Airfoil, $R = 2000$, $\alpha = 5°$, $t = 1.83$.

t = 0.118



t = 3.33

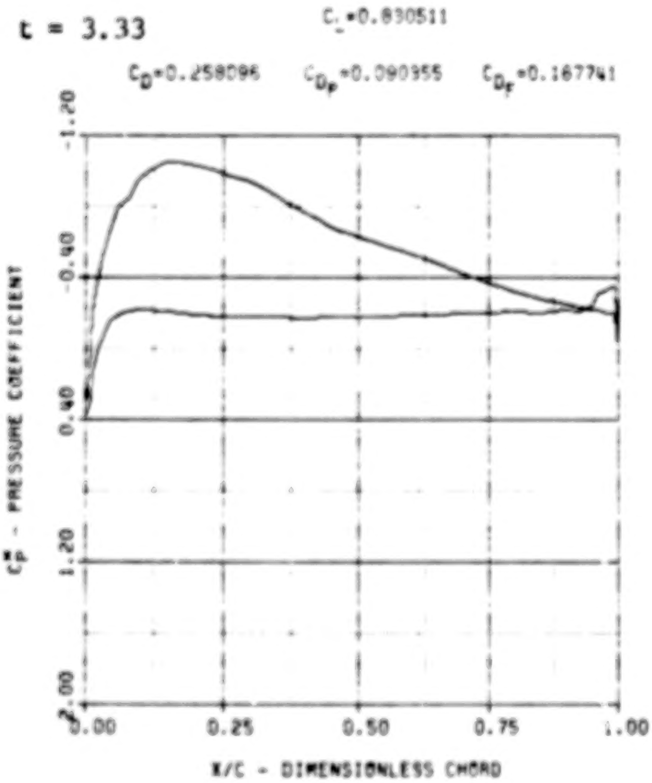


Figure 49. Pressure Distribution - Göttingen 625 Airfoil, R = 2000, α = 5°, t = 0.118 and 3.33

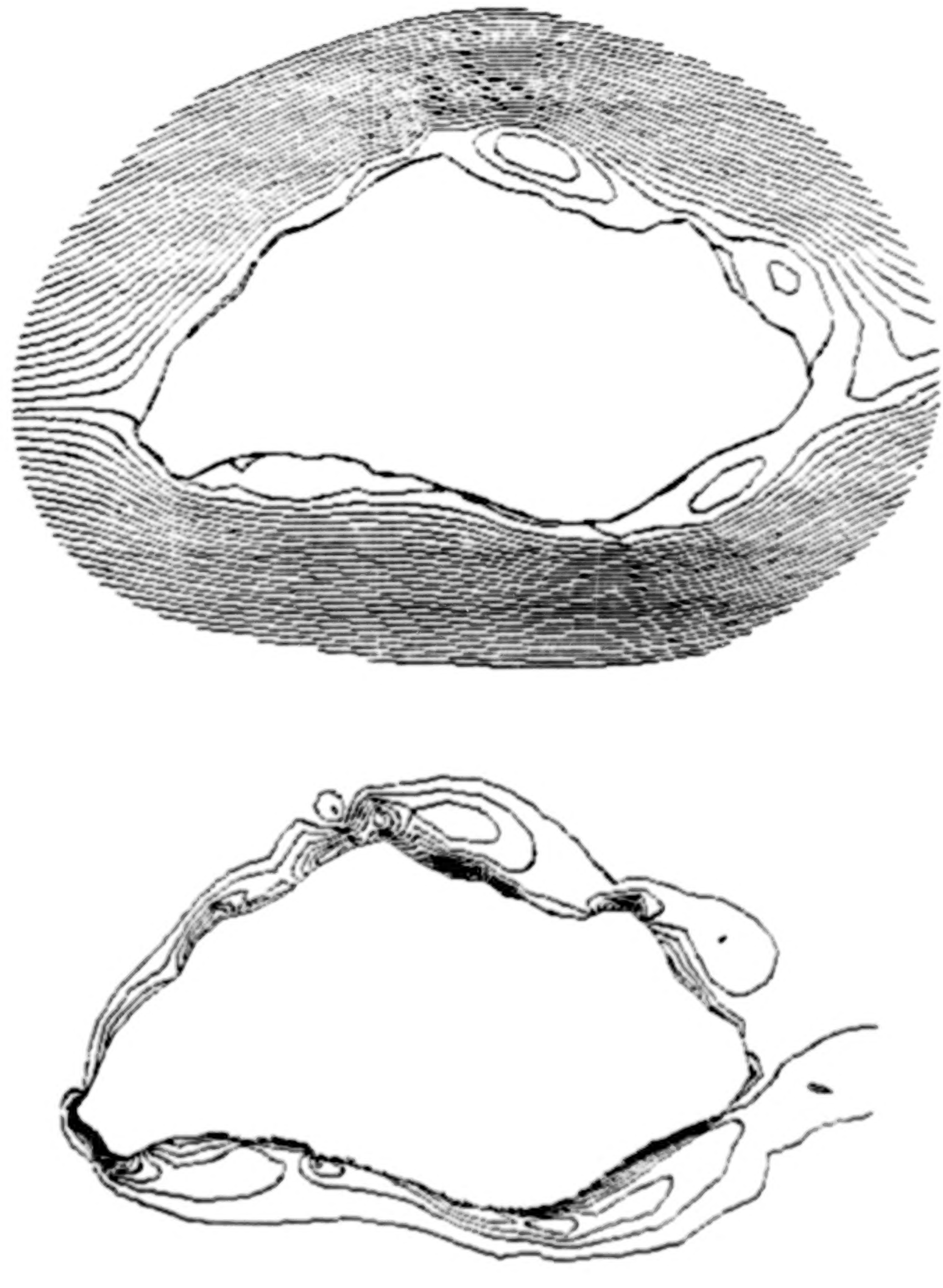


Figure 50. Streamlines and Vorticity Contours - Rock, R = 500, t = 0.5

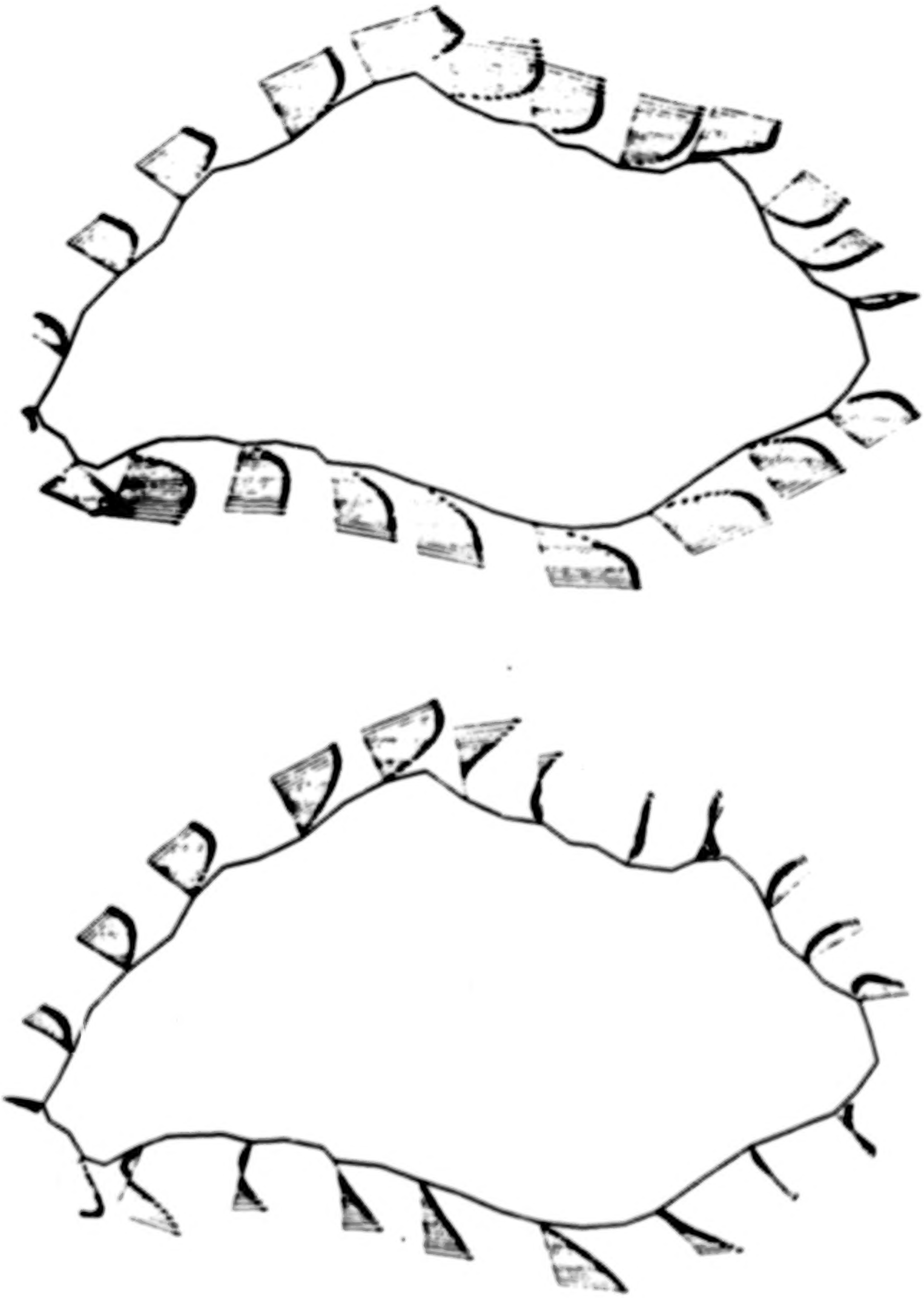
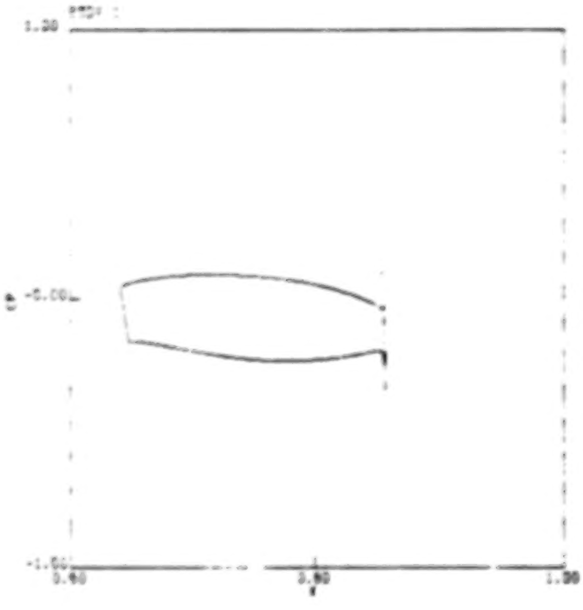


Figure 51. Velocity Profiles - Rock, R = 500, t = 1.2



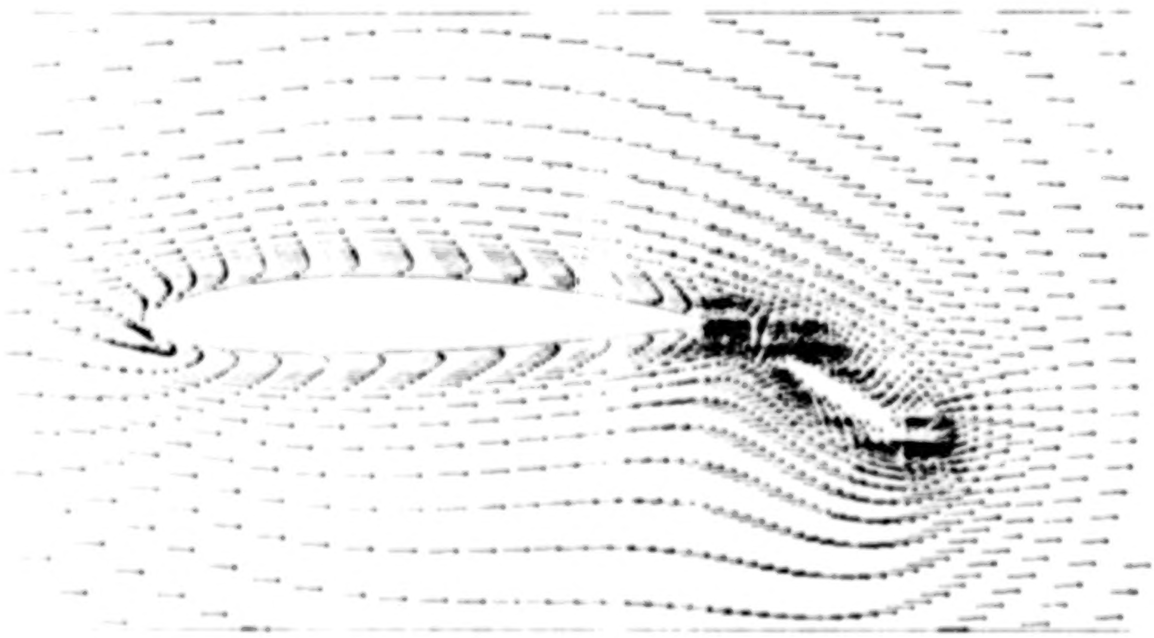


Figure 52a. Pressure Distribution and Velocity Vectors - Double Airfoil, R = 1000, t = 0.1

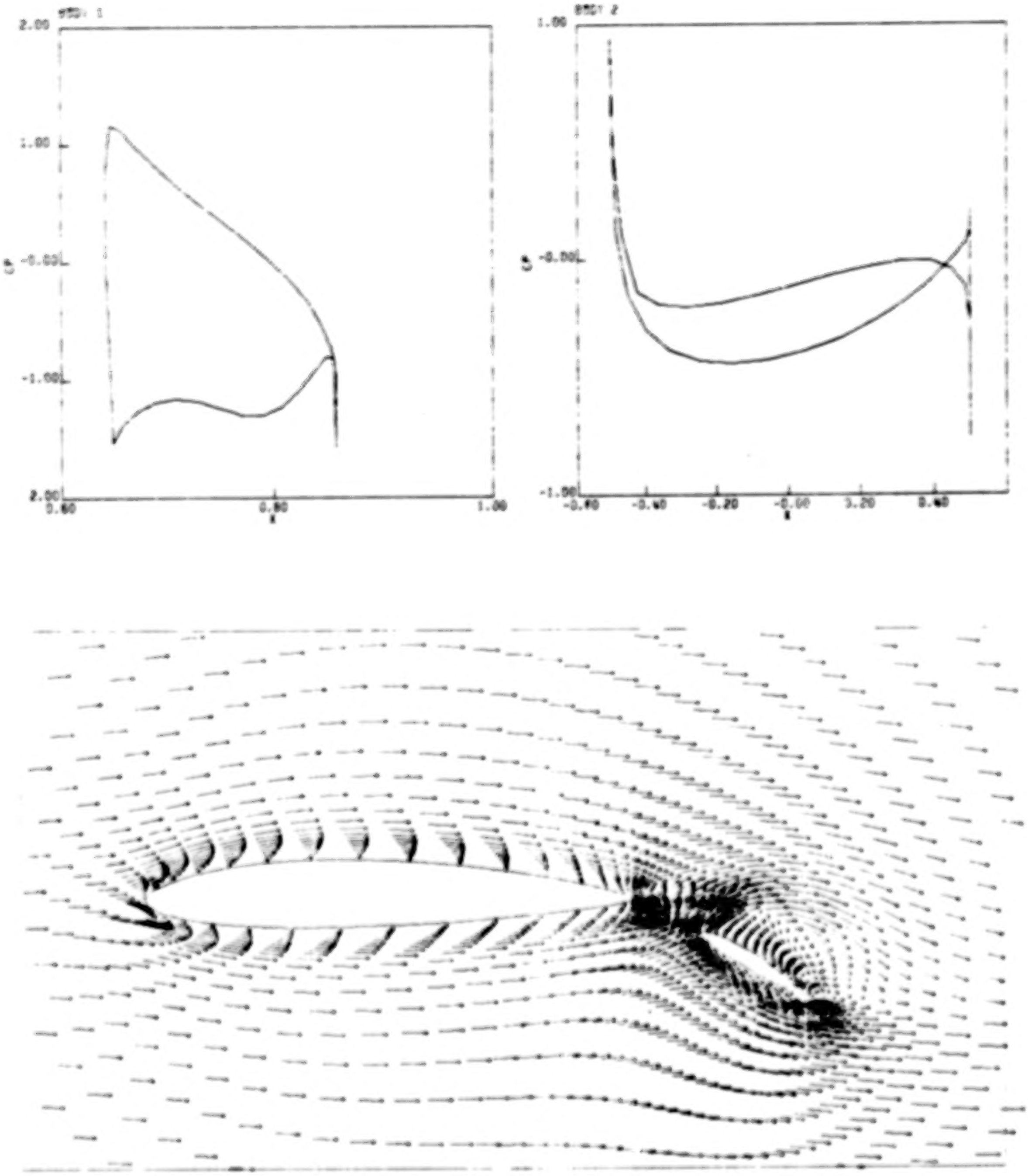


Figure 52b. Pressure Distribution and Velocity Vectors - Double Airfoil, R = 1000, t = 1.2

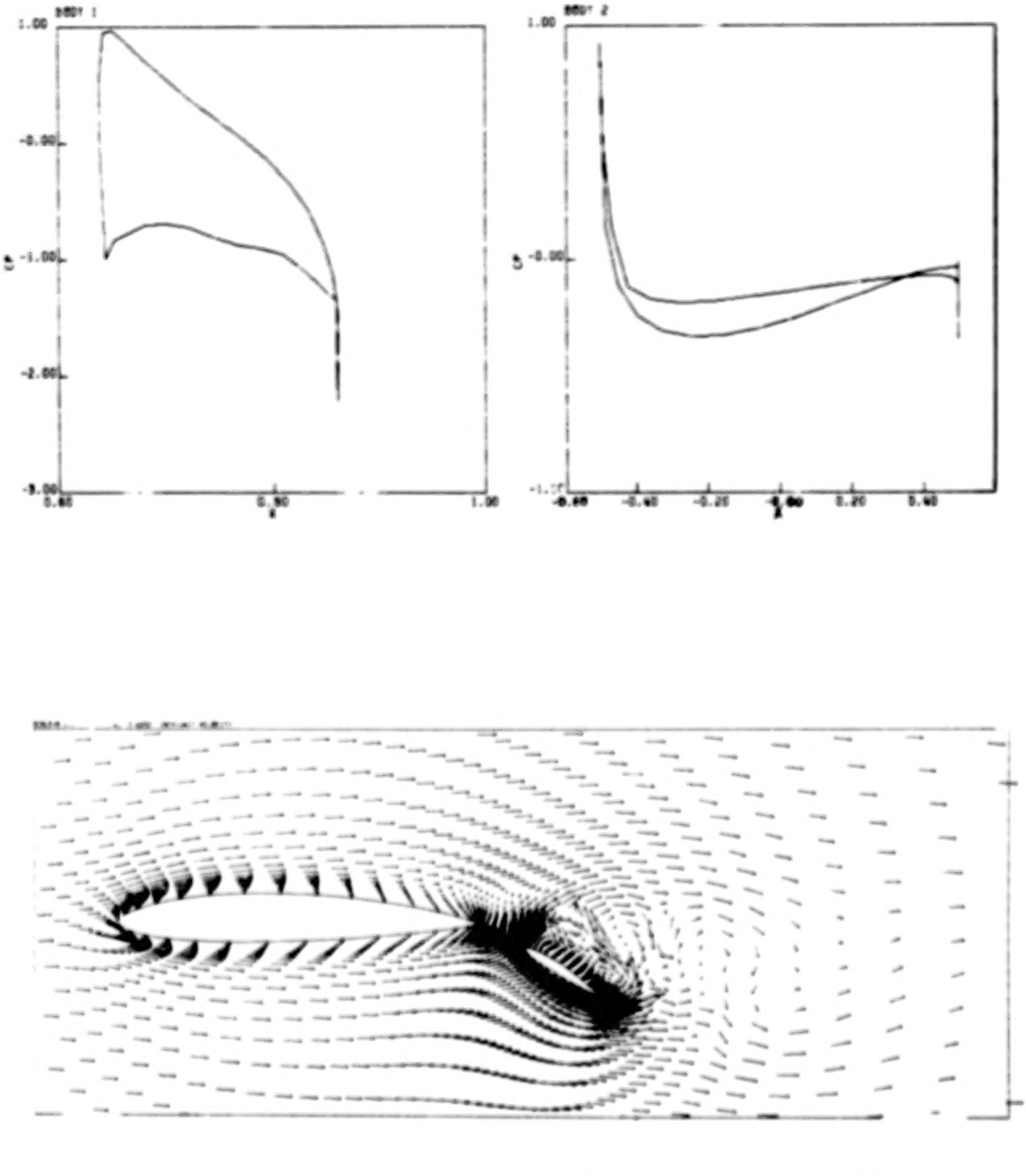


Figure 52c. Pressure Distribution and Velocity Vectors - Double Airfoil, R = 1000, t = 3.02

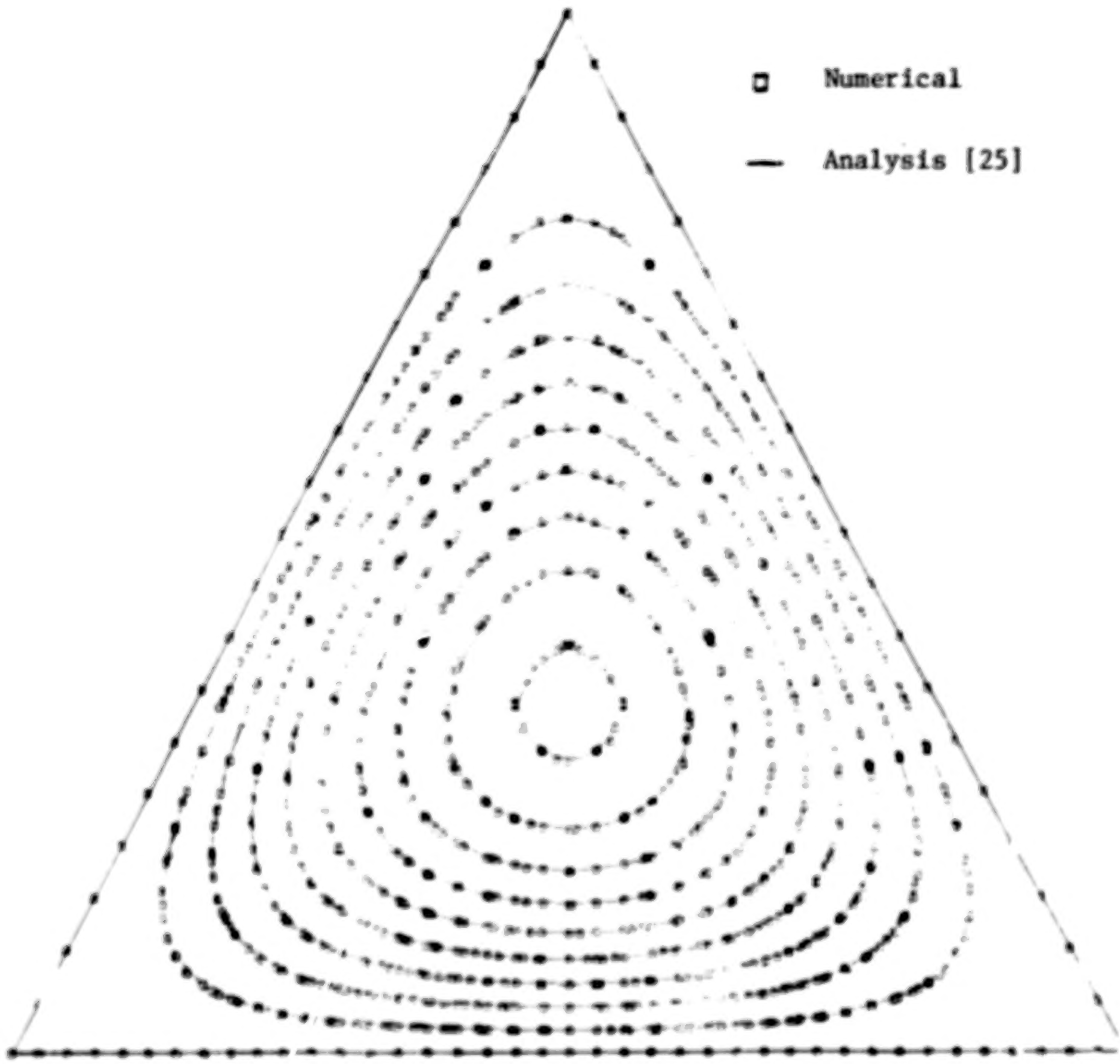


Fig. 53. Comparison of Numerical and Analytic Solution for Deflection Contours for a Simply-Supported Uniformly Loaded Plate.

APPENDIX A

DERIVATIVES AND VECTORS IN THE TRANSFORMED PLANE

This appendix contains a comprehensive set of relations in the transformed $[\xi,\eta]$ plane. A few relations involving x and y derivatives of the coordinate functions $\xi(x,y)$ and $\eta(x,y)$ are also included. Since the intent here is to provide a quick reference only, most of the algebraic development is omitted. The following function definitions are applicable throughout this appendix:

$f(x,y,t)$ - a twice continuously differentiable scalar function of x, y, and t.

$F(x,y) = \underset{\sim}{i}\,F_1(x,y) + \underset{\sim}{j}\,F_2(x,y)$ - a continuously differentiable vector-valued function of x and y. $\underset{\sim}{i}$ and $\underset{\sim}{j}$ are the conventional cartesian coordinate unit vectors.

$J = x_\xi y_\eta - x_\eta y_\xi$

$\alpha = x_\eta{}^2 + y_\eta{}^2$

$\beta = x_\xi x_\eta + y_\xi y_\eta$

$\gamma = x_\xi{}^2 + y_\xi{}^2$

$Dx = \alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta}$

$Dy = \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta}$

$\sigma = (y_\xi Dx - x_\xi Dy)/J$

$\tau = (x_\eta Dy - y_\eta Dx)/J$

Derivative Transformations

$$f_x \equiv \left(\frac{\partial f}{\partial x}\right)_{y,t} = (y_\eta f_\xi - y_\xi f_\eta)/J \tag{A.1}$$

$$f_y \equiv \left(\frac{\partial f}{\partial y}\right)_{x,t} = (x_\xi f_\eta - x_\eta f_\xi)/J \tag{A.2}$$

$$f_t \equiv \left(\frac{\partial f}{\partial t}\right)_{x,y} = \left(\frac{\partial f}{\partial t}\right)_{\xi,\eta} - \frac{1}{J}(y_\eta f_\xi - y_\xi f_\eta)\left(\frac{\partial x}{\partial t}\right)_{\xi,\eta}$$
$$- \frac{1}{J}(x_\xi f_\eta - x_\eta f_\xi)\left(\frac{\partial y}{\partial t}\right)_{\xi,\eta} \tag{A.3}$$

$$f_{xx} \equiv \left(\frac{\partial^2 f}{\partial x^2}\right)_{y,t} = (y_\eta{}^2 f_{\xi\xi} - 2y_\xi y_\eta f_{\xi\eta} + y_\xi{}^2 f_{\eta\eta})/J^2$$
$$+ [(y_\eta{}^2 y_{\xi\xi} - 2y_\xi y_\eta y_{\xi\eta} + y_\xi{}^2 y_{\eta\eta})(x_\eta f_\xi - x_\xi f_\eta)$$
$$+ (y_\eta{}^2 x_{\xi\xi} - 2y_\xi y_\eta x_{\xi\eta} + y_\xi{}^2 x_{\eta\eta})(y_\xi f_\eta - y_\eta f_\xi)]/J^3 \tag{A.4}$$

$$f_{yy} \equiv \left(\frac{\partial^2 f}{\partial y^2}\right)_{x,t} = (x_\eta{}^2 f_{\xi\xi} - 2x_\xi x_\eta f_{\xi\eta} + x_\xi{}^2 f_{\eta\eta})/J^2$$
$$+ [(x_\eta{}^2 y_{\xi\xi} - 2x_\xi x_\eta y_{\xi\eta} + x_\xi{}^2 y_{\eta\eta})(x_\eta f_\xi - x_\xi f_\eta)$$
$$+ (x_\eta{}^2 x_{\xi\xi} - 2x_\xi x_\eta x_{\xi\eta} + x_\xi{}^2 x_{\eta\eta})(y_\xi f_\eta - y_\eta f_\xi)]/J^3 \tag{A.5}$$

$$f_{xy} = [(x_\xi y_\eta + x_\eta y_\xi) f_{\xi\eta} - x_\xi y_\xi f_{\eta\eta} - x_\eta y_\eta f_{\xi\xi}]/J^2$$
$$+ [x_\eta y_\eta x_{\xi\xi} - (x_\xi y_\eta + x_\eta y_\xi) x_{\xi\eta} + x_\xi y_\xi x_{\eta\eta}](y_\eta f_\xi - y_\xi f_\eta)/J^3$$
$$+ [x_\eta y_\eta y_{\xi\xi} - (x_\xi y_\eta + x_\eta y_\xi) y_{\xi\eta} + x_\xi y_\xi y_{\eta\eta}](x_\xi f_\eta - x_\eta f_\xi)/J^3 \tag{A.6}$$

<u>Derivatives of $\xi(x,y)$ and $\eta(x,y)$</u>

$$\xi_x = y_\eta/J \tag{A.7}$$

$$\xi_y = -x_\eta/J \tag{A.8}$$

$$\eta_x = -y_\xi/J \tag{A.9}$$

$$\eta_y = x_\xi/J \tag{A.10}$$

$$\xi_{xx} = (\xi_x y_{\xi\eta} + \eta_x y_{\eta\eta})/J - (\xi_x{}^2 J_\xi + \xi_x \eta_x J_\eta)/J \tag{A.11}$$

$$\xi_{yy} = -(\eta_y x_{\eta\eta} + \xi_y x_{\xi\eta})/J - (\xi_y \eta_y J_\eta + \xi_y{}^2 J_\xi)/J \tag{A.12}$$

$$\xi_{xy} = (\eta_y y_{\eta\eta} + \xi_y y_{\xi\eta})/J - (\xi_x \xi_y J_\xi + \xi_x \eta_x J_\eta)/J \tag{A.13}$$

$$\eta_{xx} = -(\xi_x y_{\xi\xi} + \eta_x y_{\xi\eta})/J - (\xi_x \eta_x J_\xi + \eta_x{}^2 J_\eta)/J \tag{A.14}$$

$$\eta_{yy} = (\eta_y x_{\xi\eta} + \xi_y x_{\xi\xi})/J - (\xi_y \eta_y J_\xi + \eta_y{}^2 J_\eta)/J \tag{A.15}$$

$$\eta_{xy} = -(\eta_y y_{\xi\eta} + \xi_y y_{\xi\xi})/J - (\eta_x \eta_y J_\eta + \xi_y \eta_x J_\xi)/J \tag{A.16}$$

Vector Derivative Transformations

Laplacian:

$$\nabla^2 f = (\alpha f_{\xi\xi} - 2\beta f_{\xi\eta} + \gamma f_{\eta\eta})/J^2 + [(\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta})(y_\xi f_\eta - y_\eta f_\xi)$$

$$+ (\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta})(x_\eta f_\xi - x_\xi f_\eta)]/J^3 \qquad (A.17)$$

or,

$$\nabla^2 f = (\alpha f_{\xi\xi} - 2\beta f_{\xi\eta} + \gamma f_{\eta\eta} + \sigma f_\eta + \tau f_\xi)/J^2 \qquad (A.18)$$

Gradient:

$$\underset{\sim}{\nabla} f = [(y_\eta f_\xi - y_\xi f_\eta)\underset{\sim}{i} + (x_\xi f_\eta - x_\eta f_\xi)\underset{\sim}{j}]/J \qquad (A.19)$$

Divergence:

$$\underset{\sim}{\nabla} \cdot \underset{\sim}{F} = [y_\eta (F_1)_\xi - y_\xi (F_1)_\eta + x_\xi (F_2)_\eta - x_\eta (F_2)_\xi]/J \qquad (A.20)$$

Curl:

$$\underset{\sim}{\nabla} \times \underset{\sim}{F} = \underset{\sim}{k}[y_\eta (F_2)_\xi - y_\xi (F_2)_\eta - x_\xi (F_1)_\eta + x_\eta (F_1)_\xi]/J \qquad (A.21)$$

Unit Tangent and Unit Normal Vectors in the ξ,η Plane

In many applications components of vector valued functions either normal or tangent to a line of constant ξ or η are required. Similarly, directional derivatives in these directions are often needed to evaluate boundary conditions. These quantities may be obtained by trivial calculations if unit vectors tangent and normal to the ξ and η-lines are available. These vectors are developed below.

It is well known that the unit normal to the graph of $f(x,y)$. = constant is given by

$$\underset{\sim}{n}^{(f)} = \frac{\underset{\sim}{\nabla} f}{|\underset{\sim}{\nabla} f|}$$

Associating the coordinate function $\eta(x,y)$ with $f(x,y)$, we have

$$\underset{\sim}{n}^{(\eta)} = \frac{\underset{\sim}{\nabla}\eta}{|\underset{\sim}{\nabla}\eta|}$$

Utilizing equation (A.19) this reduces to

$$\underset{\sim}{n}^{(\eta)} = (-y_\xi \underset{\sim}{i} + x_\xi \underset{\sim}{j})/\sqrt{\gamma} \tag{A.22}$$

which is the unit vector normal to a line of constant η. In a similar manner the unit vector normal to a line of constant ξ is given by

$$\underset{\sim}{n}^{(\xi)} = \frac{\underset{\sim}{\nabla}\xi}{|\underset{\sim}{\nabla}\xi|} = (y_\eta \underset{\sim}{i} - x_\eta \underset{\sim}{j})/\sqrt{\alpha} \tag{A.23}$$

These vectors are illustrated as they appear in the physical plane

in Figure A.1 below.

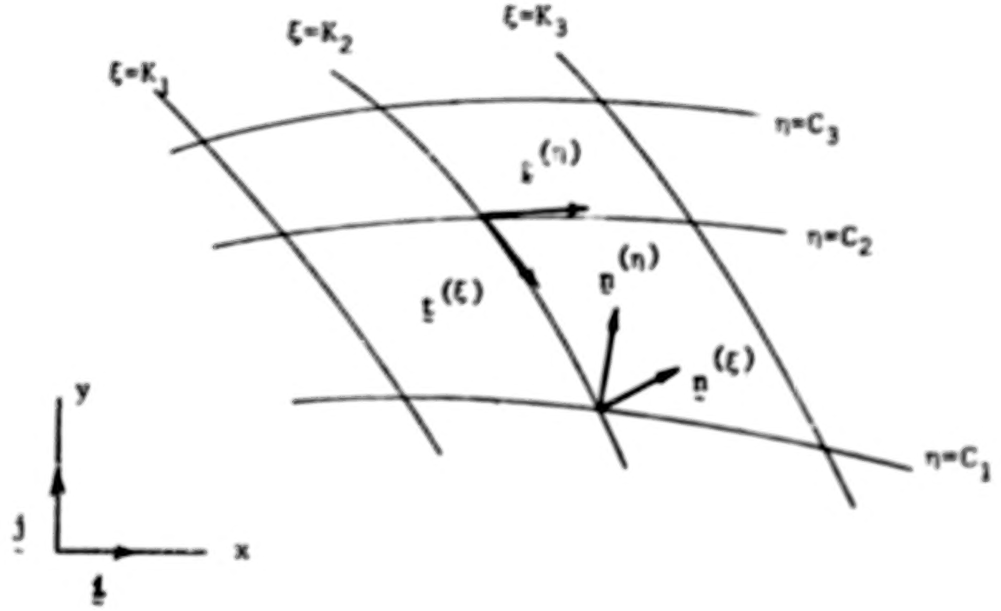


Figure A.1. Unit Tangent and Normal Vectors

The unit tangent vectors are then given by

$$\underset{\sim}{t}^{(\eta)} = \underset{\sim}{n}^{(\eta)} \times \underset{\sim}{k} = (x_\xi \underset{\sim}{i} + y_\xi \underset{\sim}{j})/\sqrt{\gamma} \tag{A.24}$$

$$\underset{\sim}{t}^{(\xi)} = \underset{\sim}{n}^{(\xi)} \times \underset{\sim}{k} = -\,(x_\eta \underset{\sim}{i} + y_\eta \underset{\sim}{j})/\sqrt{\alpha} \tag{A.25}$$

Vector Components Tangent and Normal to Lines of Constant ξ and η

$$F_{\underset{\sim}{n}}(\eta) = \underset{\sim}{n}^{(\eta)} \cdot \underset{\sim}{F} = (-y_\xi F_1 + x_\xi F_2)/\sqrt{\gamma} \tag{A.26}$$

$$F_{\underset{\sim}{t}}(\eta) = \underset{\sim}{t}^{(\eta)} \cdot \underset{\sim}{F} = (x_\xi F_1 + y_\xi F_2)/\sqrt{\gamma} \tag{A.27}$$

$$F_{\underset{\sim}{n}}(\xi) = \underset{\sim}{n}^{(\xi)} \cdot \underset{\sim}{F} = (y_\eta F_1 - x_\eta F_2)/\sqrt{\alpha} \tag{A.28}$$

$$F_{\underset{\sim}{t}}(\xi) = \underset{\sim}{t}^{(\xi)} \cdot \underset{\sim}{F} = -(x_\eta F_1 + y_\eta F_2)/\sqrt{\alpha} \tag{A.29}$$

<u>Directional</u> <u>Derivatives</u>

$$\frac{\partial f}{\partial \underset{\sim}{n}(\eta)} = \underset{\sim}{n}^{(\eta)} \cdot \underset{\sim}{\nabla} f = (\gamma f_\eta - \beta f_\xi)/J\sqrt{\gamma} \tag{A.30}$$

$$\frac{\partial f}{\partial \underset{\sim}{t}(\eta)} = \underset{\sim}{t}^{(\eta)} \cdot \underset{\sim}{\nabla} f = f_\xi/\sqrt{\gamma} \tag{A.31}$$

$$\frac{\partial f}{\partial \underset{\sim}{n}(\xi)} = \underset{\sim}{n}^{(\xi)} \cdot \underset{\sim}{\nabla} f = (\alpha f_\xi - \beta f_\eta)/J\sqrt{\alpha} \tag{A.32}$$

$$\frac{\partial f}{\partial \underset{\sim}{t}(\xi)} = \underset{\sim}{t}^{(\xi)} \cdot \underset{\sim}{\nabla} f = - f_\eta/\sqrt{\alpha} \tag{A.33}$$

<u>Integral</u> <u>Transform</u>

Scalar Function:

$$\int_D f(x,y)dxdy = \int_{D^*} f(x(\xi,\eta)\ ,\ y(\xi,\eta))\,|J|\,d\xi d\eta \tag{A.34}$$

Vector Function:

Consider a vector integral in the physical plane of the form

$$\underset{\sim}{I} = \int_S f(x,y)\ \underset{\sim}{n}(x,y)\ dS \tag{A.35}$$

where S is the closed cylindrical surface of unit depth whose perimeter is specified by the contour Γ_1 in the physical plane (see Figure A.2) and whose outward unit normal at any point is given by $\underset{\sim}{n}(x,y)$.

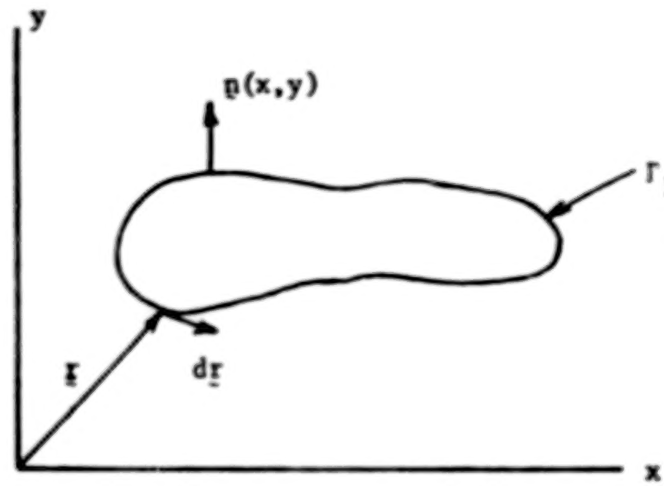


Figure A.2. Integration Around Contour Γ_1

If $r = r(x,y)$ is the position vector describing Γ_1, then

$$dS = (1.0)\ |d\underset{\sim}{r}|$$

which implies that (A.35) becomes

$$\underset{\sim}{I} = \oint_{\Gamma_1} f(x,y)\ \underset{\sim}{n}(x,y)\ |d\underset{\sim}{r}| \tag{A.36}$$

We now wish to transform (A.36) to the (ξ,η) plane. Consider $|d\underset{\sim}{r}|$:

$$|d\underset{\sim}{r}| = \sqrt{(dx)^2 + (dy)^2} = \sqrt{\gamma(d\xi)^2 + \beta d\xi d\eta + \alpha(d\eta)^2}$$

$$= \sqrt{\gamma}\ d\xi \tag{A.37}$$

since Γ_1 transforms to $\Gamma_1{}^*$, a constant η-line ($\eta = \eta_1$). Noting that $\underset{\sim}{n}(x,y) = \underset{\sim}{n}^{(\eta_1)} = (-y_\xi \underset{\sim}{i} + x_\xi \underset{\sim}{j})/\sqrt{\gamma}$ (Equation (A.22)), Equation (A.37) becomes

$$\underset{\sim}{I} = \int_{\xi min}^{\xi max} f(x(\xi,\eta_1) \ , \ y(\xi,\eta_1)) \ (x_\xi \underset{\sim}{j} - y_\xi \underset{\sim}{i}) d\xi$$

$$= \int_{\xi min}^{\xi max} f(\xi,\eta_1) \ (x_\xi \underset{\sim}{j} - y_\xi \underset{\sim}{i}) d\xi \tag{A.38}$$

where ξmin and ξmax are the minimum and maximum values respectively of ξ on $\Gamma_1{}^*$. Note that all quantities in (A.38) are evaluated along $\eta = \eta_1$. If the vector $\underset{\sim}{n}(x,y)$ is incorporated into the function $f(x,y)$, we can define the vector function $\underset{\sim}{f}(x,y)$ as

$$\underset{\sim}{f}(x,y) \equiv f(x,y) \ \underset{\sim}{n}(x,y)$$

Equation (A.36) now becomes

$$\underset{\sim}{I} = \int_{\xi min}^{\xi max} \underset{\sim}{f}(\xi,\eta_1) \ \sqrt{\gamma} \ d\xi \tag{A.39}$$

which is merely an alternate form of (A.38).

APPENDIX B

PRESERVATION OF EQUATION TYPE

When coordinate transformations are utilized as an aid in solving a given partial differential equation, it is imperative that the equation not change type under the transformation. Such an invariance will now be demonstrated for the transformation discussed in Section II. Consider the general, second order, quasi-linear partial differential equation

$$A(x,y,f)f_{xx} + B(x,y,f)f_{xy} + C(x,y,f)f_{yy} + E(x,y,f)f_{x}$$

$$+ F(x,y,f)f_{y} + G(x,y,f) = 0 \qquad \text{(B.1)}$$

where $f = f(x,y)$ is a twice continuously differentiable scalar function and A, B, C, E, F, and G are continuous functions. Recall that the equation type is determined by the coefficient functions A, B, and C as follows:

Elliptic if $B^2 - 4AC < 0$

Parabolic if $B^2 - 4AC = 0$

Hyperbolic if $B^2 - 4AC > 0$

Utilizing equations (A.1), (A.2), (A.4) - (A.6), and (A.7) - (A.10) of Appendix A, equation (B.1) transforms to

$$A^*f_{\xi\xi} + B^*f_{\xi\eta} + C^*f_{\eta\eta} + E^*f_{\xi} + F^*f_{\eta} + G^* = 0 \qquad \text{(B.2)}$$

where:

$$A^* \equiv A\xi_x^2 + B\xi_x\xi_y + C\xi_y^2$$

$$B^* \equiv 2A\xi_x\eta_x + B(\xi_x\eta_y + \xi_y\eta_x) + 2C\xi_y\eta_y$$

$$C^* \equiv A\eta_x^2 + B\eta_x\eta_y + C\eta_y^2$$

$$E^* \equiv A\xi_{xx} + B\xi_{xy} + C\xi_{yy} + E\xi_x + F\xi_y$$

$$F^* \equiv A\eta_{xx} + B\eta_{xy} + C\eta_{yy} + E\eta_x + F\eta_y$$

$$F^* \equiv F$$

Now, consider $(B^*)^2 - 4A^*C^*$:

$$\begin{aligned}(B^*)^2 - 4A^*C^* &= [2A\xi_x\eta_x + B(\xi_x\eta_y + \eta_x\xi_y) + 2C\xi_y\eta_y]^2 \\ &\quad - 4(A\xi_x^2 + B\xi_x\xi_y + C\xi_y^2)(A\eta_x^2 + B\eta_x\eta_y + C\eta_y^2) \\ &= (B^2 - 4AC)(\xi_x\eta_y - \xi_y\eta_x)^2 \\ &= (B^2 - 4AC)/J^2\end{aligned}$$

Since $J^2 > 0$, $B^2 - 4AC$ and $(B^*)^2 - 4A^*C^*$ are either both positive, both negative, or both zero. This implies that (B.1) and (B.2) have the same type.

APPENDIX C

PRESERVATION OF INTEGRAL CONSERVATION RELATIONS IN THE TRANSFORMED PLANE

It is now shown that the divergence property is not lost in the transformed plane. Let $\underset{\sim}{F}(x,y)$ be a vector valued function defined on the physical plane (Region D in Figure 1) whose component functions, $F_1(x,y)$ and $F_2(x,y)$, are continuously differentiable on D. Let $S(x,y)$ be a continuously differentiable function on D and suppose $\underset{\sim}{F}(x,y)$ and $S(x,y)$ are related by the partial differential equation

$$\underset{\sim}{\nabla} \cdot \underset{\sim}{F} = S \tag{C.1}$$

Integrating (C.1) over an area RCD, we obtain

$$\int_R [(F_1)_x + (F_2)_y]dxdy = \int_R Sdxdy$$

Application of Green's Theorem to the integral on the left yields the conservation relation

$$\oint_C (F_1dy - F_2dx) = \int_R Sdxdy \tag{C.2}$$

where C is the boundary curve of Region R taken in the proper direction. In a physical sense the line integral represents a flux through the boundary of R, and the integral over R a source within R.

Now, in the transformed plane (C.1) becomes

$$\frac{1}{J}\{[y_\eta(F_1)_\xi - y_\xi(F_1)_\eta] + [x_\xi(F_2)_\eta - x_\eta(F_2)_\xi]\} = S \quad \text{(C.3)}$$

But also note

$$\begin{aligned}(y_\eta F_1 - x_\eta F_2)_\xi + (x_\xi F_2 - y_\xi F_1)_\eta = {} & [y_\eta(F_1)_\xi - y_\xi(F_1)_\eta] \\ & + [x_\xi(F_2)_\eta - x_\eta(F_2)_\xi] + [y_{\xi\eta}F_1 - x_{\xi\eta}F_2] \\ & + [x_{\xi\eta}F_2 - y_{\xi\eta}F_1]\end{aligned}$$

which, since the last two terms cancel, implies that (C.3) becomes:

$$(y_\eta F_1 - x_\eta F_2)_\xi + (x_\xi F_2 - y_\xi F_1)_\eta = SJ$$

Integrating over the area R* in the transformed plane and applying Green's Theorem as before yields the relation

$$\oint_{C^*} [(y_\eta F_1 - x_\eta F_2)d\eta - (x_\xi F_2 - y_\xi F_1)d\xi] = \int_{R^*} SJd\xi d\eta \qquad \text{(C.4)}$$

where C* is the boundary curve of R* (R* and C* are the images of R and C respectively). To see that (C.4) expresses the conservation relation in the transformed plane parallel to that expressed by (C.2) in the physical plane, consider calculating the components of $\underset{\sim}{F}$ normal to lines of constant ξ and η. Utilizing equations (A.26) and (A.28) we have

$$\underset{\sim}{F} \cdot \underset{\sim}{n}^{(\eta)} = (x_\xi F_2 - y_\xi F_1)/\sqrt{\gamma}$$

$$\underset{\sim}{F} \cdot \underset{\sim}{n}^{(\xi)} = (y_\eta F_1 - x_\eta F_2)/\sqrt{\alpha}$$

Let $\underset{\sim}{r}$ be the position vector describing the points along the curve C*. Then, utilizing conventional techniques,

$$|d\underset{\sim}{r}| = \sqrt{\gamma(d\xi)^2 + \beta d\xi d\eta + \alpha(d\eta)^2}$$

Along a line of constant η we have

$$|d\underset{\sim}{r}|_\eta = \sqrt{\gamma}\, d\xi$$

and along a line of constant ξ

$$|d\underset{\sim}{r}|_\xi = \sqrt{\alpha}\, d\eta$$

Thus, the flux across a line of constant η is

$$\underset{\sim}{F} \cdot \underset{\sim}{n}^{(\eta)} |d\underset{\sim}{r}|_\eta = (x_\xi F_2 - y_\xi F_1) d\xi$$

and across a line of constant ξ becomes

$$\underset{\sim}{F} \cdot \underset{\sim}{n}^{(\xi)} |d\underset{\sim}{r}|_\xi = (y_\eta F_1 - x_\eta F_2) d\eta$$

These are the relations appearing in the flux terms of (C.4). The flux terms thus have an exact analog to those appearing in (C.2). Hence, the conservative relation (C.4) in the transformed plane expresses conservation in the physical plane over the non-square area formed by intersection of the curvilinear ξ and η coordinate lines in a manner which is precisely equivalent to the conservation expressed by (C.2) over the square area formed by the intersection of the x and y coordinate lines.

APPENDIX D

FINITE DIFFERENCE APPROXIMATIONS IN THE TRANSFORMED PLANE

This appendix contains a compilation of the second order finite difference expressions used to approximate partial derivatives in the transformed plane. Computational molecules for the derivative approximations appear to the right of each expression given. Combined difference forms for the transformation parameters α, β, γ, J, σ, and τ are also included here. Since the field step size is immaterial in the (ξ,η) plane, it is taken as unity for all approximations and does not appear explicitly. The following definitions are used throughout this appendix:

$f=f(\xi,\eta)$ - a twice continuously differentiable function of ξ and η.

$x=x(\xi,\eta)$ - coordinate transformation function defined by Equation (13).

$y=y(\xi,\eta)$ - coordinate transformation function defined by Equation (13).

α, β, γ, J, σ, and τ - transformation parameters defined in Appendix A.

In addition the following notational convention is utilized to indicate the position at which functions and derivatives are evaluated in the discrete (ξ,η) plane:

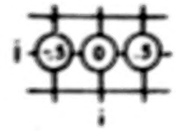
$$f_{i,j} \equiv f(\xi_i,\eta_j)\ ,(f_\xi)_{i,j} \equiv f_\xi(\xi_i,\eta_j),\ (f_\eta)_{i,j} \equiv f_\eta(\xi_i,\eta_j)$$

$$(f_{\xi\xi})_{i,j} \equiv f_{\xi\xi}(\xi_i,\eta_j)\ ,(f_{\xi\eta})_{i,j} \equiv f_{\xi\eta}(\xi_i,\eta_j)\ ,\ (f_{\eta\eta})_{i,j} \equiv f_{\eta\eta}(\xi_i,\eta_j)$$

<u>Derivative</u> <u>Approximations</u>

First derivative, central differences:

$$(f_\xi)_{i,j} = (f'_\xi)_{i,j}/2$$

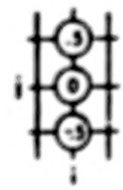


(D.1a)

where

$$(f'_\xi)_{i,j} \equiv f_{i+1,j} - f_{i-1,j} \tag{D.1b}$$

$$(f_\eta)_{i,j} = (f'_\eta)_{i,j}/2$$



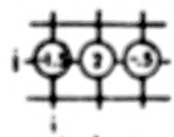
(D.2a)

where

$$(f'_\eta)_{i,j} \equiv f_{i,j+1} - f_{i,j-1} \tag{D.2b}$$

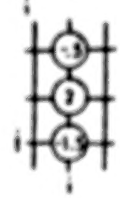
First derivative, forward differences:

$$(f_\xi)_{i,j} = (-f_{i+2,j} + 4f_{i+1,j} - 3f_{i,j})/2$$



(D.3)

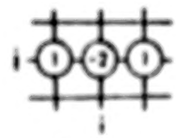
$$(f_\eta)_{i,j} = (-f_{i,j+2} + 4f_{i,j+1} - 3f_{i,j})/2$$



(D.4)

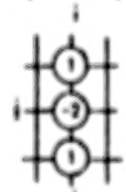
Second derivative, central differences:

$$(f_{\xi\xi})_{i,j} = f_{i+1,j} - 2f_{i,j} + f_{i-1,j} \equiv (f'_{\xi\xi})_{i,j}$$



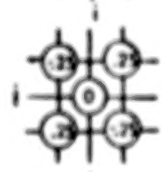
(D.5)

$$(f_{\eta\eta})_{i,j} = f_{i,j+1} - 2f_{i,j} + f_{i,j-1} \equiv (f'_{\eta\eta})_{i,j}$$



(D.6)

$$(f_{\xi\eta})_{i,j} = (f'_{\xi\eta})_{i,j}/4$$



(D.7a)

where

$$(f'_{\xi\eta})_{i,j} \quad f_{i+1,j+1} - f_{i+1,j-1} + f_{i-1,j-1} - f_{i-1,j+1} \tag{D.7b}$$

<u>Transformation Parameters</u>

$$\alpha_{i,j} = \alpha'_{i,j}/4 \tag{D.8a}$$

where

$$\alpha'_{i,j} \equiv (x'_\eta)^2_{i,j} + (y'_\eta)^2_{i,j} \tag{D.8b}$$

$$\beta_{i,j} = \beta'_{i,j}/4 \tag{D.9a}$$

where

$$\beta'_{i,j} \equiv (x'_\xi)_{i,j}(x'_\eta)_{i,j} + (y'_\xi)_{i,j}(y'_\eta)_{i,j} \tag{D.9b}$$

$$\gamma_{i,j} = \gamma'_{i,j}/4 \tag{D.10a}$$

where

$$\gamma'_{i,j} \equiv (x'_\xi)^2_{i,j} + (y'_\xi)^2_{i,j} \tag{D.10b}$$

$$J_{i,j} = J'_{i,j}/4 \tag{D.11a}$$

where

$$J'_{i,j} \equiv (x'_\xi)_{i,j}(y'_\eta)_{i,j} - (x'_\eta)_{i,j}(y'_\xi)_{i,j} \tag{D.11b}$$

$$\sigma_{i,j} = \sigma'_{i,j}/2 \tag{D.12a}$$

where

$$\sigma'_{i,j} \equiv [(y'_\xi)_{i,j}(Dx')_{i,j} - (x'_\xi)_{i,j}(Dy')_{i,j}]/J'_{i,j} \tag{D.12b}$$

$$\tau_{i,j} = \tau'_{i,j}/2 \tag{D.13a}$$

where

$$\tau'_{i,j} \equiv [(x'_\eta)_{i,j}(Dy')_{i,j} - (y'_\eta)_{i,j}(Dx')_{i,j}]/J'_{i,j} \tag{D.13b}$$

and where

$$(Dx')_{i,j} \equiv \alpha'_{i,j}(x'_{\xi\xi})_{i,j} - 2\beta'_{i,j}(x'_{\xi\eta})_{i,j} + \gamma'_{i,j}(x'_{\eta\eta})_{i,j} \tag{D.14a}$$

$$(Dy')_{i,j} \quad \alpha'_{i,j}(y'_{\xi\xi})_{i,j} - 2\beta'_{i,j}(y'_{\xi\eta})_{i,j} + \gamma'_{i,j}(y'_{\eta\eta})_{i,j} \tag{D.14b}$$

APPENDIX E

CONTOUR PLOTS

This appendix presents a detailed discussion of the methods used to determine contour plots of a function defined in the ζ,η plane. The numerical procedures which are used to transform the contour to a physical plane representation are also covered.

Determination of Contours in the ζ,η Plane

Let $\phi = \phi((\zeta,\eta))$ be a function defined in the region $\overline{D}^*$ (Figure 1) possessing continuous second derivatives. Since $\overline{D}^*$ is closed and bounded, let $m(\phi) \equiv \min\ \{\phi((\zeta,\eta)) \mid [\zeta,\eta]\epsilon\ \overline{D}^*\}$ and $M(\phi) \equiv \max\ \{\phi((\zeta,\eta)) \mid [\zeta,\eta]\epsilon\ \overline{D}^*\}$. If Φ is a number such that $m(\phi) \leq \Phi \leq M(\phi)$, then we define the Φ-contour of $\phi((\zeta,\eta))$, $C_T(\Phi)$, as the set

$$C_T(\Phi) = \{[\zeta,\eta] \mid [\zeta,\eta]\epsilon\ \overline{D}^* \text{ and } \phi((\zeta,\eta)) = \Phi\}$$

Graphically, $C_T(\Phi)$ is the curve created by the intersection of the graph of $\phi((\zeta,\eta))$ and the plane $\phi((\zeta,\eta)) = \Phi$. For plotting convenience the curve is usually projected onto the $((\zeta,\eta))$ plane. These ideas are illustrated in Figure E.1. The contour, $C_T(\Phi)$, is also often referred to as the level set of $\phi((\zeta,\eta))$ through Φ.

Now suppose that $\phi((\zeta,\eta))$ is known only in a discrete fashion. That is, let the net function $\phi_{i,j} \equiv \phi((\zeta_i,\eta_j))$ be known on the discrete set $\overline{D}^{**} = \{[\zeta_i,\eta_j] \mid \zeta_i = i\text{-}1 \text{ for } 1 \leq i \leq \text{IMAX and } \eta_j = j\text{-}1 \text{ for } 1 \leq j \leq \text{JMAX}\}$

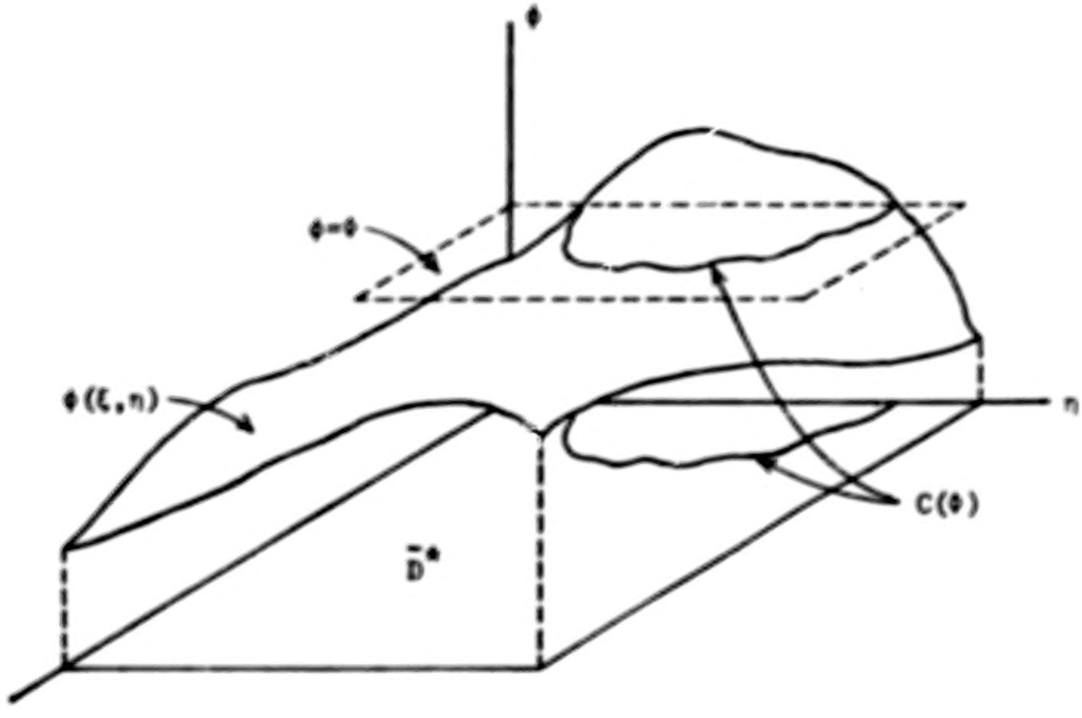


Figure E.1. Contour - Φ of $\phi(\zeta,\eta)$

(The fact that $\phi_{i,j}$ may only be an approximation to $\phi(\zeta\ ,\eta\ )$ is immaterial to the current discussion). If similar defintions for $m(\phi)$ and $M(\phi)$ are made for $\phi_{i,j}$ on the set $\bar{D}^{**}$, then the Φ-contour of $\phi_{i,j}$, denoted $C_T(\Phi)$ again, can be defined as

$$C_T(\Phi) = \{[\bar{\zeta}_k,\bar{\eta}_k] \mid 0 \leq \bar{\zeta}_k \leq \text{IMAX-1};\ 0 \leq \bar{\zeta}_k \leq \text{JMAX-1};$$

$$\bar{\phi}(\bar{\zeta}_k,\bar{\eta}_k) = \Phi;\ k=1,2,\ldots,N;\ N \geq 2\}$$

The bars over ζ_k,η_k, and ϕ are to indicate that $[\bar{\zeta}_k,\bar{\eta}_k]$ may not be an element of $\bar{D}^{**}$ and that $\bar{\phi}$ is not necessarily one of the values of the net function $\phi_{i,j}$. This is readily apparent from the discrete analog to Figure E.1 given in Figure E.2.

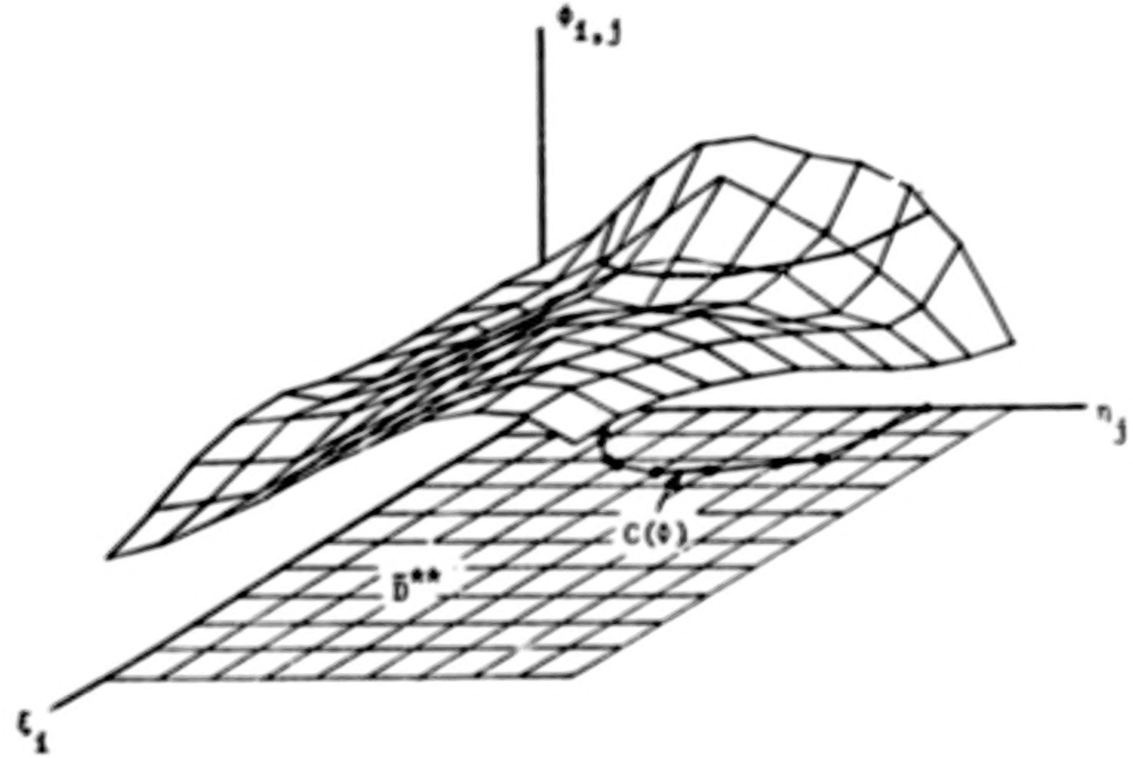


Figure E.2. Contour of $\phi(\zeta_i, \eta_j)$

Numerically, the import of the above discussion is that interpolation between the points of the discrete set $\bar{D}^{**}$ is required to determine $C_T(\phi)$. Consider a portion of grid $\bar{D}^{**}$ as shown in Figure E.3a where $1 \leq I1 < I2 \leq IMAX$ and $1 \leq J1 < J2 \leq JMAX$. Each grid block is labeled by the (i,j) coordinates of the lower left hand corner of the the block. Block (m,n) is shown on a larger scale in Figure E.3b.

In order to improve the plotted resolution consider subdividing each block into four triangles, as shown. The value of ϕ at $(m + \frac{1}{2}, n + \frac{1}{2})$ is taken as the four point average

$$\phi_{m+\frac{1}{2},n+\frac{1}{2}} = (\phi_{m,n} + \phi_{m+1,n} + \phi_{m,n+1} + \phi_{m+1,n+1})/4$$

A local (ζ,η) coordinate is affixed to each grid block as demonstrated

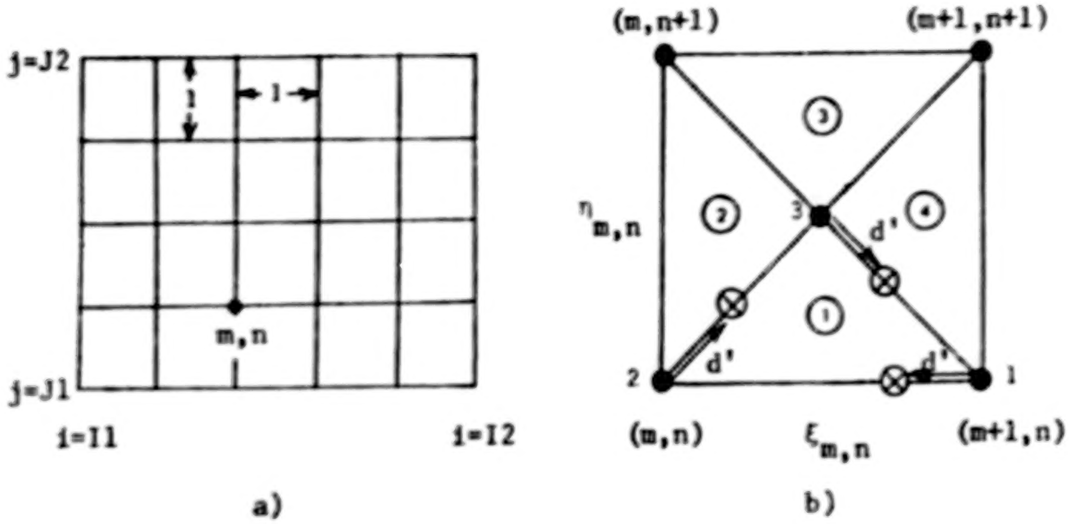


Figure E.3. Sample Grid in Set $\bar{D}^{**}$

in Figure E.3b. In order to standardize the interpolation procedures, a local (ζ,μ) coordinate system is also placed on each of the sub-triangles as illustrated in the series of drawings given in Figure E.4.

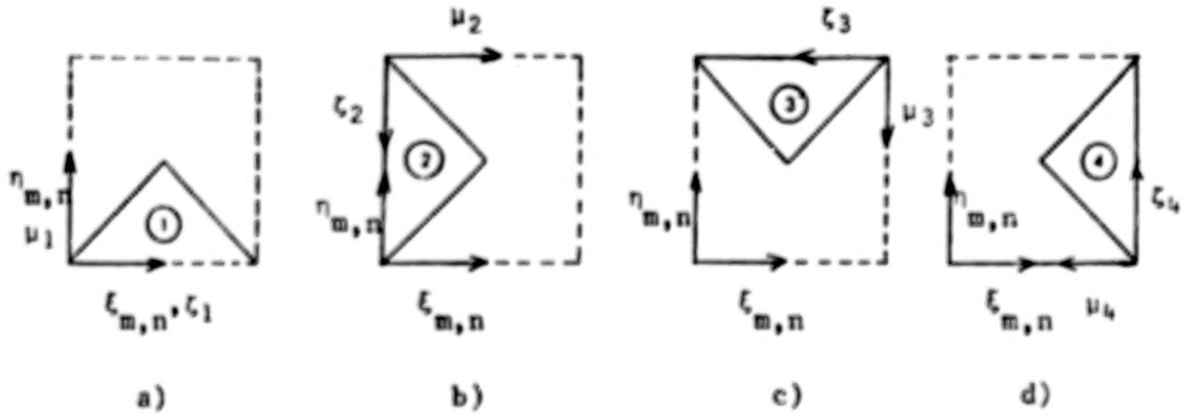


Figure E.4. Local Coordinate Systems for Triangles

Interpolation is carried out on each of the four triangles for each grid block in the segment (I1 $\leq$ i $\leq$ I2, J1 $\leq$ j $\leq$ J2) of the set $\bar{D}^{**}$ specified. In particular interpolation is performed along each of the three sides of each of the triangles if the contour value, Φ, lies between the values at the ends of the sides. Let d' be the directed distance from a given

triangle vertex to the point on the triangle side where the contour intersects that side (denoted by ⊗). This distance is illustrated in Figure E.3b for triangle ① and is defined in an analagous manner for the other triangles. If Φ_1 is the value of $\phi_{i,j}$ at a particular triangle vertex and Φ_2 the value of $\phi_{i,j}$ at the other end of the side, then d' may be expressed as

$$d' = (\text{side length})(\Phi - \Phi_1)/(\Phi_2 - \Phi_1)$$

For example, along side 1-2 of triangle ①, d' is given by

$$d'_{1-2} = (1.0)(\Phi - \phi_{m+1,n})/(\phi_{m,n} - \phi_{m+1,n})$$

Noting that the sides of the triangle have lengths 1.0, $1.0/\sqrt{2}$, and $1.0/\sqrt{2}$, the contour intersections can be expressed in the local triangle coordinates as

Side	ζ_ℓ	μ_ℓ
1-2	1-d	0
2-3	d/2	d/2
3-1	(1+d)/2	(1-d)/2

where $d \equiv (\Phi - \Phi_1)/(\Phi_2 - \Phi_1)$ and where ℓ=1,2,3, or 4 denotes the triangle number.

Once the contour intersections have been determined in the local triangle coordinates, (ζ_ℓ, μ_ℓ), they must be transformed to the grid block coordinates $(\xi_{m,n}, \eta_{m,n})$. This is done in the conventional fashion using orthogonal rotation matrices. If $[\zeta_{\ell,p}, \mu_{\ell,p}]$ are the coordinates of an intersection in triangle ℓ, then

$$\begin{bmatrix} (\xi_{m,n})_{\ell,p} \\ \\ (\eta_{m,n})_{\ell,p} \end{bmatrix} = \underline{A_\ell} \begin{bmatrix} \zeta_{\ell,p} \\ \\ \mu_{\ell,p} \end{bmatrix}$$

where

$$\underline{A_1} \equiv \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \underline{A_2} \equiv \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

$$\underline{A_3} \equiv \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad \underline{A_4} \equiv \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Note that up to three contour intersections can occur for each triangle (i.e., one on each side). Finally, the point $[(\xi_{m,n})_{\ell,p}, (\eta_{m,n})_{\ell,p}]$ is transformed to (ξ,η) coordinates by a simple linear transformation producing an element $[\bar{\xi}_k, \bar{\eta}_k]$ of the set $C_T(\Phi)$.

Transformation to the Physical Plane

Since contours in the (ξ,η) plane are of little interest, $C_T(\Phi)$ must be transformed to the physical plane. This is made possible through the use of the coordinate transformation functions $x(\xi_i,\eta_j)$ and $y(\xi_i,\eta_j)$. Again interpolation is required since almost all elements of $C_T(\Phi)$ are not elements of $\bar{D}^{**}$ on which the discrete functions $x(\xi_i,\eta_j)$ and $y(\xi_i,\eta_j)$ are defined. As illustrated in Figure E.5 this implies a double linear interpolation must be performed. If $[\bar{\xi}_k, \bar{\eta}_k]$ denotes an element of $C_T(\Phi)$, the first step is to locate the ξ and η values bracketing $\bar{\xi}_k$ and $\bar{\eta}_k$. Denoting these by ξ_i, ξ_{i+1} and η_j, η_{j+1} as shown in the figure, the values of $\bar{x}_k$ and $\bar{y}_k$ are calculated as follows

$$\bar{x}_k = (\bar{\eta}_k - \eta_j)(X_{j+1} - X_j)/(\eta_{j+1} - \eta_j) + X_j$$

where

$$X_j = (\bar{\xi}_k - \xi_i)(x_{i+1,j} - x_{i,j})/(\xi_{i+1} - \xi_i) + x_{i,j}$$

$$X_{j+1} = (\bar{\xi}_k - \xi_i)(x_{i+1,j+1} - x_{i,j+1})/(\xi_{i+1} - \xi_i) + x_{i,j+1}$$

for all k=1,2,...,N. Similar expressions are used to calculate $\bar{y}_k$.

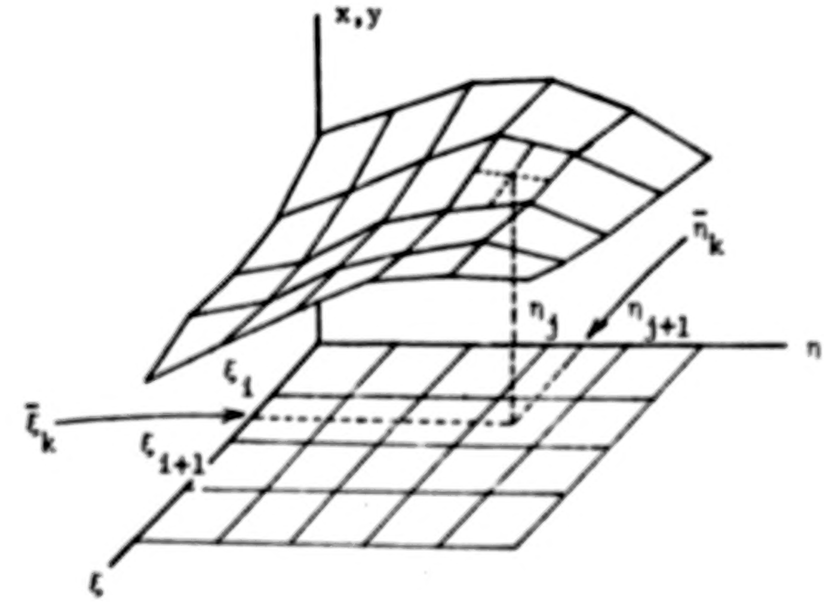


Figure E.5. Interpolation for x and y

90

50

END

DEC 4 1978